

This Page Is Inserted by IFW Operations  
and is not a part of the Official Record

## **BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning documents *will not* correct images,  
please do not report the images to the  
Image Problem Mailbox.**

**THIS PAGE BLANK (USPTO)**

特許協力条約に基づいて公開された国際出願

BNSDOCID: <WO\_9717652A1\_1\_>

(57) 要約

追記型記録媒体をあたかも書換え型記録媒体のように用いることができる情報処理装置及び情報処理方法。

記録媒体上の物理的記録位置を仮想アドレス空間上において論理アドレスと対応させて管理することにより、物理的記録位置が変更されても同一ブロックの論理アドレスを変更させることのないアドレス管理機構を実現できる。かくしてライトワンス型の記録媒体をあたかも書き換え可能な記録媒体として扱うことができる。

また記録媒体上の利用可能残容量を監視することにより、メモリ内で作成又は更新中のデータを確実に記録媒体上に記録することができる。

情報としての用途のみ

PCTに基づいて公開される国際出願をパンフレット第一頁にPCT加盟国を同定するために使用されるコード

AL	アルバニア	EE	エストニア	LR	リベリア	RU	ロシア連邦
AM	アルメニア	ES	スペイン	LS	レソト	RD	ラトヴィア
AT	オーストリア	FI	フィンランド	LT	リトアニア	SE	スウェーデン
AU	オーストラリア	FR	フランス	LU	ルクセンブルグ	SG	シンガポール
AZ	アゼルバイジャン	GB	イギリス	LV	ラトヴィア	SK	スロバキア共和国
BB	バルバドス	GE	グルジア	MC	モナコ	SN	セネガル
BE	ベルギー	GH	ガーナ	MD	モルドバ	SZ	スワジランド
BG	ブルガリア	GN	ギニア	MG	マダガスカル	TD	チャド
BJ	ベナン	GR	ギリシャ	MK	マケドニア	TG	トーゴ
BR	ブラジル	HU	ハンガリー	VI	ベトナム	TM	トルクメニスタン
BY	ベラルーシ	IE	アイルランド	ML	マリ	TR	トルコ
CA	カナダ	IT	イタリア	MN	モンゴル	TT	トリニダード・トバゴ
CC	中央アフリカ共和国	JP	日本	MR	モーリタニア	UA	ウクライナ
CF	中央アフリカ共和国	KE	ケニア	MW	マラウイ	UG	ウガンダ
CG	コンゴ	KG	キルギスタン	MX	メキシコ	US	米国
CH	スイス	KP	朝鮮民主主義人民共和国	NE	ニジェール	UZ	ウズベキスタン共和国
CI	コート・ジボアール	KR	韓国	NL	オランダ	VN	ベトナム
CM	コモロ	RZ	大韓民国	NO	ノルウェー	YU	ユーゴスラビア
CN	中国	LI	リビア	NZ	ニュージーランド		
CZ	チェコ共和国			PL	ポーランド		
DE	ドイツ			PT	ポルトガル		
DK	デンマーク			RO	ルーマニア		

## 明 細 書

## 発明の名称

情報処理装置及び情報処理方法

## 技術分野

本発明は、情報処理装置及び情報処理方法に関し、例えば追記（ライトワンス）型のディスク状記録媒体に情報を記録するようになされた情報処理装置及び情報処理方法に関する。

## 背景技術

従来、大容量のデータ記録媒体としてCD-ROM(Compact Disc-Read Only Memory)と呼ばれる読出し専用のものである。このCD-ROMに記録層を形成することによって1回だけ書込みを可能とした追記型のディスク状記録媒体（以下これをCD-R(Compact Disc-Recordable)がある。

このCD-Rは従来のCD-ROMドライブを用いて再生し得ることから、例えばハードディスクに全てのデータを記録して用意しておき、この全てのデータをCD-Rに書込むことによつて、当該書込みの完了したCD-RをCD(Compact Disc)として用いるようになされている。

ところで、かかる構成のCD-Rは1回だけ書込み可能であることから、一度書き込んだ領域のデータを同一領域に書き換えることが困難であり、この結果当該CD-Rをいわゆるフロッピーディスクのような書換え可能の記録媒体として用いることが困難であつた。

また追記型のCD-Rディスクで記録データの更新を行う場合、一旦書き込まれた領域を書き直すことが困難であることから、この場合新たな領域に更新されたデータを書き直す作業が必要となる。このような追記型記録媒体の特徴により、当該追記型のCD-Rディスクを用いてデータの更新を行おうとする場合、当該CD-Rデ

ディスクの利用可能な残容量が更新操作に応じて減っていくことになり、利用可能容量に余裕がなくなると、メモリ上で更新処理中のデータの書き込みを行うことが困難になる問題があつた。

本発明は以上の点を考慮してなされたもので、追記型媒体をあたかも書換え型記録媒体のように用いることができると共に、追記型媒体の利用可能残容量を監視し得るようになされた情報処理装置及び情報処理方法を提案しようとするものである。

#### 発明の開示

本発明は、媒体上に記録される物理位置を論理位置と対応させるとともに、物理位置及び上記論理位置の対応表を生成し、物理位置が変更されたとき論理位置と変更後の物理位置との対応関係をメモリ上で管理する管理手段と、メモリ内のファイルデータ及び対応表の要素データを書込みデータとして媒体に記録する記録手段と、媒体の書込み可能残容量を算出する算出手段と、書込み可能残容量とメモリ内の書込みデータ量とを比較する比較手段と、比較手段の比較結果が書込み量の超過を示すとき、当該超過が検出された結果を超過検出結果として出力する出力手段とを備える。

また本発明は、上記記録手段において、超過検出結果が出力されたとき、直ちにメモリ内の書込みデータを媒体上に記録する。

また本発明は、上記情報処理装置において、超過検出結果が出力されたとき、所定の表示手段に警告表示する警告手段を備える。

また本発明は、媒体上に記録される物理位置を論理位置と対応させるとともに、物理位置及び論理位置の対応表を生成し、物理位置が変更されたとき論理位置と変更後の物理位置との対応関係をメモリ上で管理する工程と、メモリ内のファイルデータ及び対応表の要素データを書込みデータとして媒体に記録する工程と、媒体の書込み可能残容量を算出する工程と、書込み可能残容量とメモリ内の書込みデータ量とを比較する工程と、比較手段の比較結果が書込み量の超過を示す

とき、当該超過が検出された結果を超過検出結果として出力する工程とを備える。

本発明によれば、媒体上の物理的記録位置をメモリ内の仮想アドレス空間上において論理アドレスと対応させて管理することにより、物理的記録位置が変更されても同一ブロックの論理アドレスを変更させることのないアドレス管理機構を実現できる。かくしてライトワンス型の記録媒体をあたかも書き換え可能な記録媒体として扱うことができる。

また追記型でなる媒体の利用可能残容量を監視することにより、メモリ内で作成又は更新中のデータを確実に記録媒体上に記録することができる。

#### 図面の簡単な説明

図 1 は、本発明による CD-R ディスク装置を全体構成を示すブロック図である。

図 2 は、本発明による CDRFS の構成を示すブロック図である。

図 3 は、シーケンスマネージャ S Q M による複数の仮想アドレス空間の管理の説明に供する略線図である。

図 4 は、B \* 木 (B Star-tree) による管理構造の説明に供する略線図である。

図 5 は、B \* 木 (B Star-tree) による管理構造の説明に供する略線図である。

図 6 は、エクステンツによる対応表を示す略線図である。

図 7 は、ノードテーブルを示す略線図である。

図 8 は、ブロック数の最大数の説明に供する略線図である。

図 9 (A) 及び図 9 (B) は、データブロックの更新の説明に供する略線図である。

図 10 は、フラッシュ動作の説明に供するフローチャートである。

図 11 (A) 乃至図 11 (D) は、CD-R ディスクの論理構造を示す略線図である。

- 図 1 2 は、P V D の構造を示す略線図である。
- 図 1 3 は、スーパーブロックの構造を示す略線図である。
- 図 1 4 は、スーパーブロックリストエントリの構造を示す略線図である。
- 図 1 5 は、タグエントリの構造を示す略線図である。
- 図 1 6 は、ノードテーブルの構造を示す略線図である。
- 図 1 7 は、B \* 木 (B Star-tree) インデックスノードの構造を示す略線図である。
- 図 1 8 は、インデックスレコードの構造を示す略線図である。
- 図 1 9 は、シーケンス B \* 木 (B Star-tree) リーフノードの構造を示す略線図である。
- 図 2 0 は、エクステントレコードの構造を示す略線図である。
- 図 2 1 は、ディレクトリ B \* 木 (B Star-tree) リーフノードの構造を示す略線図である。
- 図 2 2 は、ディレクトリレコードエリアの構造を示す略線図である。
- 図 2 3 は、ディレクトリレコードの構造を示す略線図である。
- 図 2 4 は、ディレクトリレコードキーの構造を示す略線図である。
- 図 2 5 は、タイプの種類の説明に供する略線図である。
- 図 2 6 は、ファイルディレクトリレコードの構造を示す略線図である。
- 図 2 7 は、ディレクトリディレクトリレコードの構造を示す略線図である。
- 図 2 8 は、リンクディレクトリレコードの構造を示す略線図である。
- 図 2 9 は、フラッシュオール動作を示すフローチャートである。
- 図 3 0 (A) 乃至図 3 0 (C) は、書き直しの動作の説明に供する略線図である。
- 図 3 1 はメディアの記録可能残量を監視する処理手順を示すフローチャートである。
- 図 3 2 は、仮想アドレス空間を各ファイルに対応させずに処理する場合の説明に供する略線図である。



発明を実施するための最良の形態

図1において、1は全体としてCD-Rディスク装置を示し、CD-RディスクDISCに対して書き込むデータ又はCD-RディスクDISCから読み出されたデータを処理する情報処理装置部4、ユーザに対して情報処理装置部4の処理データ、処理状態等を提供する陰極線管又は液晶表示装置等となる表示装置2、情報処理装置部4に対してデータを入力するキーボード等となる入力装置3及び、CD-RディスクDISCに対してデータを書込み又は読み出すCD-Rドライブ装置5を有する。

情報処理装置部4は、システム全体の動作を制御するCPU(Central Processing Unit)6、種々の情報や各種プログラムが一時記憶されるRAM(Random Access Memory)7、CPU6が動作する際に必要な基本的なプログラムが格納されているROM(Read Only Memory)8、表示装置2に情報を出力するための入出力回路(I/O)9、入力装置3からの情報を取り込むための入出力回路(I/O)10、各種プログラムが格納されたハードディスクをアクセスするためのハードディスクドライブ(HDD)11、当該ハードディスクドライブ11をアクセスするためのインタフェース回路(I/F)12、CD-Rドライブ5をアクセスするためのインタフェース回路(I/F)13をそれぞれ有する。なお、RAM7は単なるメモリ機能の他に、キャッシュバッファの機能を含むものである。

かかる構成の情報処理装置部4において、CPU6がROM8に記憶されているプログラムに基づいてハードディスク11からインタフェース(I/F)12を介してCD-R用のファイルシステム(CDRFS:Compact Disc Recordable File System)のプログラムを読み出し、これをRAM7に格納する。そしてCPU6はその読み出したCDRFSを起動することによってシステムを立ち上げる。

立ち上がったCD-Rディスク装置1において、データをCD-RディスクDISCに記録する場合、CPU6はCDRFSの管理の元にユーザが作成したデータを所定のフォーマットでブロック化し、そのデータをインタフェース回路13を介してCD

-Rドライブ5に送出するとともに、CD-Rドライブ5に対して書込み指示を与える。これを受けたCD-Rドライブ5は受け取ったデータをパケットと呼ばれる書込みのデータ単位で順次CD-RディスクDISCに記録する。

またCD-RディスクDISCに記録されたデータを読み出す場合、情報処理装置部4のCPU6は、インタフェース回路13を介してCD-Rドライブ5に読み出し指示を与える。これを受けたCD-Rドライブ5はCD-RディスクDISCをアクセスしてパケット記録されているデータを読み出し、インタフェース回路13を介してRAM7に送出する。

図2はCD-R装置1においてCD-RディスクDISCをあたかも書換え可能な記録媒体であるかのようにユーザに対して見せかけるソフトウェアSWの全体構成を示し、I/Oを介して入力されるユーザからの命令はアプリケーションソフトウェアAP及びオペレーティングシステムOSにおいて解釈され、CD-R用のファイルシステムCDRFSのファイルマネージャFLMに渡される。

ファイルシステムCDRFSは、上層部分を構成するファイルマネージャFLMと下層部分を構成する仮想デバイスマネージャIMMとによつて構成され、ファイルマネージャFLMはディレクトリ及びファイルの管理を行う。従つてオペレーティングシステムOSからファイルマネージャFLMに対して例えばファイルの上書き命令が渡されると、ファイルマネージャFLMは当該命令によつて指定されたファイル名に基づいて仮想デバイスマネージャIMMに形成された対応する仮想アドレス空間を指定する。

ここで仮想デバイスマネージャIMMは、図3に示すようにファイルマネージャFLMに対して複数の仮想アドレス空間CVx(CV1、CV2、CV3、…)を提供するものであり、各仮想アドレス空間CVxはそれぞれ単数又は複数のデータブロックからなるデータブロック列によつて構成される。このデータブロック列はシーケンスと呼ばれ、ファイルマネージャFLMが管理する各ファイルに対応する。従つてファイルマネージャFLMは、仮想デバイスマネージャIMMに対してシーケンス番号を指定することにより目的とするファイルの仮想ア

ドレス空間  $CV_x$  を指定する。このときファイルマネージャ  $FLM$  はシーケンスキー  $SQK$  (図2) と呼ばれる 64 ビットの論理アドレスによつて各仮想アドレス空間  $CV_x$  をブロック単位で指定することができる。

すなわち仮想デバイスマネージャ  $IMM$  は、シーケンスキー  $SQK$  によつて各仮想アドレス空間  $CV_x$  をブロック単位で管理されており、シーケンスキー  $SQK$  の上位 32 ビットは仮想アドレス空間  $CV_x$  固有のシーケンス番号が割り当てられ、下位 32 ビットは、上位 32 ビットで指定されたシーケンス (仮想アドレス空間) 内のブロック  $BLK$  (図3) を指定するシーケンスブロック番号である。従つて当該シーケンスブロック番号によつて管理し得るブロック数 (すなわち各仮想アドレス空間  $CV_x$  が持つことのできるブロック数) は、1 つの仮想アドレス空間 (1 つのシーケンス) につき  $2^{32}$  ブロックである。各ブロック  $BLK$  は CD-R ディスクフォーマットに合わせて 2048 バイトであり、これにより仮想デバイスマネージャ  $IMM$  は 1 つのシーケンス (すなわち 1 ファイル) として 8 テラバイトを上限とする大きさのものを管理できる。

このように、仮想デバイスマネージャ  $IMM$  は、各仮想アドレス空間  $CV_x$  をファイルに対応させて設けることにより、各仮想アドレス空間  $CV_x$  がファイルそのものとなる。従つて仮想デバイスマネージャ  $IMM$  はファイルが指定されると、当該ファイルの位置を論理アドレスに変換して探索するといった煩雑な手順を実行することなく直ちにファイルへのアクセスを実行できる。

かくしてファイルマネージャ  $FLM$  から 64 ビットのシーケンスキー  $SQK$  で表されるファイルに直接対応した論理アドレスが仮想デバイスマネージャ  $IMM$  に渡されると、当該仮想デバイスマネージャ  $IMM$  のシーケンスマネージャ  $SQM$  (図2) は、 $B^*$  木 ( $B$  Star-tree) と呼ばれる多分木 (Multiway tree) による探索手法を用いてシーケンスキー  $SQK$  で表される論理アドレスと CD-R ディスク上の物理アドレスとを対応させる。

すなわちシーケンスマネージャ  $SQM$  の  $B^*$  木 ( $B$  Star-tree) は、図4に示すようにツリー構造を有し、中間ノード (分岐) としてのインデックスノード  $D$

と、論理アドレス及び物理アドレスの対応を示すエクステント (EXT x) を有するリーフノード E、F、G とによつて構成される。

各リーフノード E、F 及び G は、シーケンスキー S Q K によつて表される論理アドレスと物理アドレス L B A との関係を表すエクステント EXT x をそれぞれ単数又は複数個だけシーケンスキー S Q K の昇順に格納している。すなわちエクステント EXT x は、CD-R ディスク上の物理位置に連続して並ぶブロックのうち、シーケンスキー S Q K が昇順に連続するブロック列を 1 つの単位として管理 (表現) するものであり、当該エクステント EXT x は、当該エクステント EXT x によつて管理される連続した物理ブロックの先頭ブロックのシーケンスキー S Q K と当該シーケンスキー S Q K に対応する物理アドレス L B A とレングス (Length) とからなる。レングス (Length) はそのレングスが含まれるエクステントの物理アドレス L B A を先頭にして当該エクステント EXT x によつて表現される連続した物理ブロック数を表す。従つて例えば、エクステント EXT x が (0, 0 56 5) で表される場合、0, 0 というシーケンスキー S Q K (論理アドレス) に対応する CD-R ディスク上の物理アドレス L B A が 56 であり、当該エクステントによつて表されるデータが CD-R ディスク上において物理アドレス L B A (=56) を先頭にして 5 ブロック連続していることを表している。

この 1 つのエクステント EXT x に管理される 5 つのデータブロックは、CD-R ディスク上において連続した物理領域に書かれることから、連続した物理アドレス L B A に同一ファイル (シーケンス) のデータを記録することにより、論理アドレスと物理アドレス L B A とのアドレス変換表の要素となるエクステント EXT x の数が増大することを回避し得る。実際に、同一ファイルのデータは連続して処理される確率が高い点に着目すると、CD-R ディスク上の物理位置に連続したシーケンスキー S Q K のブロックが連続して書かれることになり、このようなデータをエクステント EXT x によつて管理することにより、シーケンスマネージャ S Q M の管理構造を構成する要素であるエクステント EXT x そのものの数を減らすことができる。例えば、エクステント EXT x が (0, 0 56 5) である場

合、当該エクステント E X T x によつて表されるディスク上の物理位置に同じシーケンス番号（ファイル）のデータ（例えば 2 ブロック）が続くように追記される場合には、エクステント E X T x が (0, 0 56 7) となつて当該管理データとしてのエクステント E X T x は増加しない。

図 4 において、B \* 木 (B Star-tree) の中間ノードを構成するインデックスノード D には、その枝を構成する各リーフノード E、F 及び G の各先頭のエクステント情報 E X T x のシーケンスキー S Q K (key1、key2、key3、……) が対応するリーフノード E、F 又は G のノード番号と共に格納され、当該シーケンスキー S Q K (key1、key2、key3、……) が指定されると、ノード番号によつて対応付けられたリーフノード E、F 又は G がノードテーブル (図 5) の参照によつて CD-R ディスク上の物理アドレス L B A から読み出される。

従つてシーケンスマネージャ S Q M は、シーケンスキー S Q K が指定されると、インデックスノード D から当該シーケンスキー S Q K が含まれる範囲の先頭キー（シーケンスキー S Q K）を探索する。例えば目的とするシーケンスキー S Q K がインデックスノード D に格納されている第 1 のシーケンスキー key1 と第 2 のシーケンスキー key2 との間の値であるとする、シーケンスマネージャ S Q M は、第 1 のシーケンスキー S Q K key1 を有するエクステント E X T 1 1 を先頭に有するリーフノード E を選択し、このリーフノード E 内を順次探索して行く。このように各リーフノード E、F 及び G 内の複数のエクステント E X T x が、そのシーケンス番号の昇順に並んでいることにより、B \* 木 (B Star-tree) の手法を用いて容易に所望のシーケンスキー S Q K で指定されたデータ列の物理アドレス L B A を探索することができる。

因みに図 5 は実際の B \* 木 (B Star-tree) による探索方法を示し、CD-R ディスク上に記録されているスーパーブロック (Super Block (S V D と呼ぶ)) A、ノードテーブル (Node Table) B、インデックスノード (Index node) D、リーフノード (Leaf node) E、F、G から目的とする CD-R ディスク上の物理アドレス L B A (H、I、J、……) を探索する。すなわちシーケンスマネージャ S Q M

は、スーパーブロックに記録されているノードテーブルBの物理アドレスLBAに基づいてCD-Rディスク上のノードテーブルBを参照すると共に、スーパーブロックに記録されているルートノード番号によつて指定されるノードテーブルBのノード番号からインデックスノードDの物理アドレスLBAを割り出す。これによりシーケンスマネージャSQMは、CD-Rディスク上のインデックスノードDを参照することができ、図4について上述したように、このインデックスノードDにおいて所望のシーケンスキーSQKに対応するノード番号を読み出し、当該ノード番号に対応するリーフノードE、F又はGの物理アドレスLBAをノードテーブルBにおいて読み出す。これにより、CD-Rディスク上から目的とするリーフノードE、F又はGを読み出すことができ、当該リーフノードE、F又はGを参照することによつて、このとき指定されているシーケンスキーSQKに対応するエクステントEXTxを読み出すことができる。このエクステントEXTxによつて目的とするデータブロック列のCD-Rディスク上の位置(H、I、J、……)が求まる。

因みに図6はシーケンス番号と物理アドレスLBAとの対応関係の一例を示し、これら4つの対応関係は、シーケンス番号(123456781h)、LBA(1000h)、レンジ(5)で表される1つのエクステントEXTxによつて表現することができる。B\*木(B Star-tree)を構成する各管理データもCD-Rディスク上に記録されることにより、内容の更新に伴う書き換えの必要が生じる。従つてB\*木(B Star-tree)を構成するデータブロックについても論理アドレスと物理アドレスLBAとの対応表と、その管理機構が設けられる。CDRFSにおいてはB\*木(B Star-tree)を構成するブロックに対して図5について上述したようなノード番号(Node Number)と呼ばれる32ビットの論理アドレスを付け、当該論理アドレスと物理アドレスLBAとの対応表を論理アドレスを添字にした配列テーブルとして管理している。

図7はノードテーブルの一例を示し、このノードテーブルからノード番号「0」のB\*木(B Star-tree)ブロックは物理アドレスLBAが「30」の位置に

記録されていることが分かる。ノードテーブル配列テーブル以外に 2 つのデータが付随している。すなわち「Number of entry」は配列の要素数を示し、「free」は配列内での未使用要素リストの先頭を示す。未使用要素リストとは、配列内で使われていない要素の管理機構であり、配列要素の再利用を簡単にすることができる。未使用要素の中身にはブロックアドレスの代わりに次の未使用要素の添字が入っている。最後の未使用要素の中身には配列の要素数「Number of entry」が入っている。この例ではtable2が未使用要素リストの先頭で、以下table4、table1となる。

ノードテーブルはCD-Rディスク上の連続領域に記録することによつて当該ノードテーブルを管理する機構を省略し得るようになされている。B\*木(B Star-tree)に変更が生じて当該B\*木(B Star-tree)を構成するブロックを書き換えた場合には、更新したノードテーブルをCD-Rディスクに書き込む。例えば1ギガバイトのデータを管理する際に必要なB\*木(B Star-tree)のブロック数の最大値は、各エクステンツEXTxが1ブロックを参照し、各ノードが半分しか使用されていないときを考えると良いので、図8に示すように、8095ブロックになる。B\*木(B Star-tree)を構成するブロック1つにつき、論理アドレスを添字にした配列テーブルに4バイト必要であることから、テーブルの大きさとしては、次式、

$$\begin{aligned} 8095\text{block} \times 4\text{ byte} + 2 \times 4\text{ byte} \\ = 32388\text{byte} = 15.8\text{block} \approx 16\text{block} \quad \cdots \cdots (1) \end{aligned}$$

に示すように、16ブロックとなり、1ギガ空間を管理する場合でもB\*木(B Star-tree)の管理テーブルとして必要な連続領域は最大で16ブロックであることが分かる。CD-RFSでは、CD-Rディスク上に記録する記録単位として32ブロックからなるパケットを用いており、最大16ブロックのノードテーブルは1パケット内に収められる。従つてCD-RFSでは、後述するフラッシュ動作時において

CD-Rディスクに書き込む最後のパケットにその他の管理情報とともにこのノードテーブルを格納する。

このようにしてCDRFS においては、データブロックの論理、物理アドレス管理機構はB\*木(B Star-tree)とノードテーブルとによる二重構造となる。このような二重構造にする理由は、ノードテーブルのように単純な配列テーブルによる論理、物理アドレス管理機構だけを用いると、表示に大きな連続領域が必要になる点にある。例えば上述の例と同様に1ギガ空間を管理する場合を考えると、次式、

$$1 \times 2^{20} \text{kbyte} / 2 \text{kbyte} \times 4 / 2048 \text{kbyte} = 1024 \text{block} \quad \dots\dots (2)$$

に示すように1024ブロックもの連続領域が必要となる。またB\*木(B Star-tree)のみで管理しようとする、B\*木(B Star-tree)のノードを物理アドレスLBAによつて参照することになり、B\*木(B Star-tree)のノードの更新の度に参照の書き換えが他のB\*木(B Star-tree)のノードに波及することになる。従つてCDRFS では、論理、物理アドレス管理機構としてB\*木(B Star-tree)とノードテーブルの二重構造としている。

かくしてシーケンスマネージャSQMにおいて、図4及び図5について上述した探索手法(B\*木(B Star-tree))によつて論理アドレスであるシーケンスキーSQKからそのシーケンスが記録されている物理アドレスLBAが求められると、当該物理アドレスLBAはそのシーケンスキーSQKとともに図2に示すキャツシヤマネージャCAMに渡される。

キャツシヤマネージャCAMは、指定された物理アドレスLBAに対応したデータブロックのCD-Rディスクからの読み出し及び書込みをキャツシユブロックと呼ばれるキャツシユバッファを介して行う。すなわちデータを書き換えるために指定された物理アドレスLBA及びそのシーケンスキーSQKが渡されると、キャツシユマネージャCAMは当該指定された物理アドレスLBAで示されるデー



タブロックがすでにキャツシユバッファ内に存在するか否かを判断する。ここで否定結果が得られると、キャツシユマネージャCAMは当該データブロックをCD-Rディスクから読み出してキャツシユバッファ内のブロック（キャツシユブロック）に格納し、当該格納されたデータブロックに仮の物理アドレス（Temporary LBA）を割り振る。この仮の物理アドレス（Temporary LBA）は、CD-Rディスク上に書き込み位置が確定していないキャツシユ内のデータブロックに割り当てられるものである。このように実際に書き込みが完了するまでキャツシユマネージャCAMによつて仮の物理アドレス（Temporary LBA）でアドレス管理することにより、シーケンスマネージャSQMは物理アドレスLBAの位置が実であるか又は仮であるかを意識することなくデータブロックにアクセスすることができる。

またこのときキャツシユマネージャCAMは、当該データブロックに対するシーケンスキーSQK（シーケンスマネージャSQMから渡されたもの）をキャツシユバッファ管理表に記憶して、当該ブロックデータが格納されたキャツシユブロックに対するデータポイントと仮の物理アドレスTemporary LBAとをシーケンスマネージャSQMに返す。シーケンスマネージャSQMは当該仮の物理アドレスTemporary LBAとシーケンスキーSQKとの対応関係をB\*木（B Star-tree）に登録しておく。

これと同時に、キャツシユマネージャCAMから返されたキャツシユブロックのデータポイントはシーケンスマネージャSQMからさらにファイルマネージャFLMに返され、ここでユーザの指定したデータの更新が行われる。このようにして書き換えられたブロックはキャツシユマネージャCAMに渡され、高速メモリで構成されたライトキャツシユブロック（Write Cache Block）においてダーティブロックと呼ばれる更新済ブロックとして管理される。このときダーティブロックはキャツシユバッファ内においてのみ更新又は作成された状態であり、未だCD-Rディスクに記録されていない。従つてキャツシユマネージャCAMは当該ダーティブロックが所定数（32ブロック）に達すると、これを1パケットとしてデバイスドライバを介してCD-RディスクDISCに書き込む。

このようにデータブロックが1パケット分溜まるまでライトキヤツシュブロック内に残すことにより、当該データがCD-Rディスク上に書き込まれる前に再び同一データブロックに対する更新が実行された場合にキヤツシュバッファ内のデータの書換えだけを行えば良く、CD-Rディスク上に対する更新（すなわち新たな物理アドレスLBAへの書込み）を回避し得る。

以上の説明は、既にCD-Rディスク上に存在するデータブロックを書き換えるための処理であるが、ファイルマネージャFLMは仮想デバイスマネージャIMM（図2）による仮想アドレス空間だけを見ながらデータの書き換えを行う。従って記録メディアとしてライトワンス型のCD-Rディスクが用いられていても、ファイルマネージャFLMは、仮想アドレス空間においてデータの書き換えを、通常書換え可能型メディアに対する操作と同じように行うことができる。

これに対して、データブロックを新たに生成する場合、シーケンスマネージャSQMはキヤツシュマネージャCAMに対して生成しようとするブロックのシーケンスキーSQKを渡してブロックの生成を要求する。キヤツシュマネージャCAMはキヤツシュ内にブロックを割り当て、これに仮の物理アドレス（Temporarily LBA）を割り振ってシーケンスマネージャSQMに返す。このときシーケンスマネージャSQMからキヤツシュマネージャCAMに渡されたシーケンスキーSQKはキヤツシュマネージャCAM内のキヤツシュバッファ管理表に記憶され、実際のCD-Rディスク上への書込み動作時に用いられる。シーケンスマネージャSQMは生成しようとするブロックのシーケンスキーSQKとキヤツシュマネージャCAMから返された物理アドレスLBAとの対応関係をB\*木（B Star-tree）でなる管理構造に登録する。

またこれに対して、データブロックの消去を行う場合、シーケンスマネージャSQMはB\*木（B Star-tree）から指定したシーケンスキーSQKの物理アドレスLBAを求め、キヤツシュマネージャCAMに対して消去を指示する。キヤツシュマネージャCAMは指定された物理アドレスLBAのブロックがキヤツシュバッファ内に存在しない場合はなにも処理を行わない。これに対して当該物理

アドレス L B A のブロックがキャッシュバッファ内に存在する場合、キャッシュマネージャ C A M は当該キャッシュブロック（すなわちキャッシュマネージャ C A M が管理しているキャッシュバッファ内のブロック）を無効にする。そして最後にシーケンスマネージャ S Q M が B \* 木（B Star-tree）から削除対象のシーケンスキー S Q K のエントリを削除して消去を完了する。

ここでキャッシュ内のライトキャッシュブロックのデータを CD-R ディスク上に書き込む動作をフラッシュと呼び、第 1 にシーケンスマネージャ S Q M がファイルマネージャ F L M からフラッシュ要求を受けたとき、又は第 2 にシーケンスマネージャ S Q M がキャッシュマネージャ C A M から書き込み要求を受けたときに当該フラッシュ動作が実行される。

第 1 の場合は、アプリケーションからのフラッシュ要求や CD-R ディスクの挿入時、システムの終了時等であり CDRFS からみて外部的要因をきっかけとするものである。これに対して第 2 の場合は、CDRFS の内部的要因をきっかけとするものであり、キャッシュ内の再利用可能なブロックが CDRFS の動作を保証する上で必要となる最低数に満たない場合、又は CDRFS の信頼性を確保する上で CD-R ディスクへの書き込み動作が必要とされる場合である。

キャッシュ内の再利用可能なブロックが CDRFS の動作を保証する上で必要となる最低数に満たない場合とは、シーケンスマネージャ S Q M の基本動作であるシーケンスブロックの読み込み、生成、更新又は削除の動作を行うために十分な再利用可能ブロックがキャッシュバッファ内に確保されていない場合である。従って十分な再利用可能ブロックが確保されていない場合は、フラッシュ動作によって当該再利用可能ブロックが確保される。因みに再利用可能ブロックとは、ライトキャッシュブロックのデータが CD-R ディスク上に書き込まれることによって当該ライトキャッシュブロックが他のデータブロックのために利用可能となったもの（リードキャッシュブロックと呼ぶ）と、元々有効なデータが格納されていないキャッシュブロック（フリーキャッシュブロック：Free Cache Block と呼ぶ）の総称である。

ここでシーケンスマネージャ S Q M の動作に必要な再利用可能ブロック数について説明する。すなわちシーケンスブロックを読む際には、対象となるデータブロックと、これらのブロックが存在する物理アドレス L B A を調べるための B \* 木 (B Star-tree) 管理構造を格納するための再利用可能ブロックがキャッシュバッファ内に必要となる。この B \* 木 (B Star-tree) ブロックは検索時に 2 つ以上同時に参照されることがなく、これにより 1 つの再利用可能ブロックがあれば良い。従ってシーケンスブロック読み出し処理に必要なキャッシュバッファ内の再利用可能ブロック数の最大値  $RBC_{max}$  は、本来読み出そうとしたデータブロックの数を  $N$  とすると、次式、

$$RBC_{max}(N) = N + 1 \quad \dots\dots (3)$$

となる。

シーケンスブロックの生成、更新作業の際、CDRFS はキャッシュバッファ内に未書込みブロックを割り当てる。従って生成、更新対象となるブロックの数を満足させるために再利用可能ブロックが必要となる。実際にはシーケンスマネージャ S Q M は B \* 木 (B Star-tree) の更新を行うため、B \* 木 (B Star-tree) のノードを格納するためのデータブロックの生成、更新が発生し、生成、更新の対象となるデータブロック数以上の再利用可能ブロックがキャッシュバッファ内に必要となることがある。また、削除作業の際にも B \* 木 (B Star-tree) の更新に伴いデータブロックの生成が行われる可能性があり、この場合にも再利用可能ブロックが必要となる。

またシーケンスブロックの生成の際には、エクステント E X T<sub>x</sub> が生成され、B \* 木 (B Star-tree) に挿入される。このとき挿入先のリーフノードに余裕がないと、当該リーフノードの分割が起きて、新たなリーフノード用にデータブロックが 1 つ生成される。さらに、当該分割のために生成されたリーフノードをインデックスノードに挿入する際、挿入先のインデックスノードに余裕がないと、

当該インデックスノードを分割するために新たなデータブロックの生成が生じる。B \* 木 (B Star-tree) の深さが「3」の場合、1つのエクステント E X T x が管理するブロック数が1つであるような最も効率の低い状態を考えると、このとき管理し得るデータの総容量は、ルート以外の全てのノードが1/2 だけ書き込まれている状態において、次式、

$$170 \times 170/2 \times 145/2 \times 2 = 2095250 \approx 2 \text{ [G byte]} \quad \dots\dots (4)$$

となる。従つて容量が 700メガバイトに満たないCD-Rディスクにおいては、深さ「3」のB \* 木 (B Star-tree) のルートノードが分割されることはなく、これによりB \* 木 (B Star-tree) に対して最初のエクステント E X T x が挿入されることに伴つて生成されるデータブロック数は多くとも2つとなる。

また分割直後のインデックスノードは、最大でも 2/3までしか埋まつていないことから、当該インデックスノードを満杯にするためには最低でも新たに  $170/3 = 56$  個のインデックスの挿入が必要となる。また、隣合うインデックスノードが満杯でない場合には、分割の代わりに互いのインデックスの数を平均するための移動が行われる。従つてB \* 木 (B Star-tree) に対する最初のインデックスノードの挿入によつてインデックスノードの分割が生じてから次の分割を生じるまでに必要なB \* 木 (B Star-tree) へのインデックスノードの挿入の最低回数は  $170/3 \times 2 = 113$  となる。またリーフノードの場合も同様にして、最低  $145/3 \times 2 = 96$  個のエクステント E X T x の挿入が次の分割を引き起こすために必要となる。かくしてシーケンスマネージャ S Q M に対する1回のブロック生成呼び出しで生成されるデータブロック数の最大値  $CBC_{max}$  は、本来生成しようとするデータブロック数をNとして、次式、

$$CBC_{max}(N) = N + 2 + (N/96)/113 + (N/96) \quad \dots\dots (5)$$

となる。

これに対してデータブロックの更新の場合は、既に存在しているエクステント E X T x を取り除く作業以外は生成の場合と同様である。すなわちエクステント E X T x は連続した複数のブロックのシーケンスキー S Q K と物理アドレス L B A との対応関係を管理していることにより、その一部が取り除かれた場合、結果としてエクステント E X T x の数が 1 つ増加する場合がある。例えば図 9 (A) に示すようなエクステントが存在する場合、シーケンスキー「3」から「6」までのデータブロックを更新するために当該データブロックをエクステントから取り除くと、このエクステントは図 9 (B) に示すように、2 つのエクステントに置き換わる。このように既に存在しているエクステントの一部を取り除く際に 1 つ余分にエクステントが発生する場合がある。これを考慮してシーケンスマネージャ S Q M に対する 1 回のブロック更新呼び出しで生成されるデータブロック数の最大値  $MBC_{max}$  は、本来更新しようとするデータブロック数を N として、次式、

$$MBC_{max}(N) = N + 2 + ((N + 1)/96)/113 + ((N + 1)/96) \dots\dots (6)$$

となる。

またデータブロックの削除では、エクステントの一部削除が発生したとき、リーフノードが分割する可能性がある。従つてシーケンスマネージャ S Q M に対する 1 回のブロック削除呼び出しで生成されるデータブロック数の最大値  $DBC_{max}$  は、削除しようとするデータブロック数に関わりなく、次式、

$$DBC_{max} = 2 \dots\dots (7)$$

となる。

また一度の操作で扱うデータの大きさが決まっていない場合は、(3) 式～(7) 式のような複雑な計算が必要でない場合もある。例えば「Windows95」(商

品名) の場合「GetDiskInfo」コールでファイルシステムからそのアロケーションユニット(Allocation Unit)を読み出すと、以降当該アロケーションユニット単位でファイルシステムにアクセスするようになる。従つてアロケーションユニットを例えば32ブロックにすれば、シーケンスマネージャSQMが一度に操作しなくてはならないデータブロック数が1パケット以下となる。従つて次式、

$$DBC_{max} < CBC_{max} (32) = MBC_{max} (32) = 34 \text{ (block)} \quad \dots\dots (8)$$

となり、シーケンスマネージャSQMの動作前に34個の再利用可能ブロックがキャッシュバッファ内にあれば良い。

ライトキャッシュブロックのデータをCD-RディスクDISC上に書き込むフラッシュ(Flash)動作を説明する。図10に示すようにシーケンスマネージャSQMがキャッシュマネージャCAMに対してキャッシュバッファ内のライトキャッシュブロックをnパケット分集めるように要求する。この場合のパケット数nはキャッシュマネージャCAMの設定に依存する。

キャッシュマネージャCAMは図10のステップSP1において当該要求に応じ、ライトキャッシュブロックの中から所定の優先度にしたがってCD-RディスクDISC上に書き込むブロックのリストを作成する。この優先度は、該当するブロックへのアクセス頻度が少ないもの(最も過去にアクセスされたブロック)から順に優先させるいわゆるLRU(Least Recently Used)アルゴリズムを用いて決められており、当該優先度の高い順にキャッシュアウト(CD-Rディスク上へ書き込む)するブロックが決定される。ここでパケットに空きが生じた場合、キャッシュマネージャCAMはダミーブロックを当該空き領域に割り当てる。

キャッシュマネージャCAMは、このようにして選出されたライトキャッシュブロックに対して、ステップSP2においてキャッシュ管理表を参照して各ライトキャッシュブロックに対応するシーケンスキーSKが昇順となるようにライトキャッシュブロックのブロックリストを整列させ、続くステップSP3にお

いて、当該ブロックリストの先頭から書込み開始物理アドレス  $LBA$ 、書込み開始物理アドレス  $LBA + 1$ 、のように各ブロックが書き込まれる  $CD-R$  ディスク  $DISC$  上の予定物理アドレス（仮の実アドレス（Contact LBA）と呼ばれる）を割り振る。この予定物理アドレス（Contact LBA）は書込みが正常に終了したとき  $CD-R$  ディスク上に確定した実アドレス（Real LBA）となり、書込みが失敗したとき上述の仮の物理アドレス（Temporary LBA）となる。

かくしてステップ  $SP3$  において、各ライトキャッシュブロックに対してそのシーケンスキー  $SK$  の昇順に書込み位置（予定物理アドレス（Contact LBA））を割り当てることにより、 $CD-R$  ディスク上に連続領域に同一シーケンス（ファイル）のブロックが並ぶ確率が高くなり、この結果  $CD-R$  ディスク上から読み出す際に連続して読み出すことができることにより、読み出し性能を向上し得るとともに、同一シーケンス（ファイル）の物理的連続ブロックを 1 つのエクステント  $EXTx$  で管理する  $B$ \*木（ $B$  Star-tree）の要素数（エクステント  $EXTx$  の数）の増大を防止し得る。

かくしてステップ  $SP3$  においてライトキャッシュブロックに予定物理アドレス（Contact LBA）が割り振られると、シーケンスマネージャ  $SQM$  はキャッシュマネージャ  $CAM$  から当該割り当てられたシーケンスキー  $SK$  及び予定物理アドレス（Contact LBA）の対応表を受け取り、ステップ  $SP4$  において  $B$ \*木（ $B$  Star-tree）の更新を行う。すなわち当該ステップ  $SP4$  においてシーケンスマネージャ  $SQM$  は、予めデータ更新の際に仮の物理アドレス（Temporary LBA）が割り当てられて作成されたエクステント  $EXTx$  の当該仮の物理アドレス（Temporary LBA）を予定物理アドレス（Contact LBA：仮の実アドレス）に置き換える。予定物理アドレス（Contact LBA）はシーケンスキー  $SK$  が昇順になるように割り振られていることにより、このような更新を複数行うといくつかのエクステント  $EXTx$  がまとめられて  $B$ \*木（ $B$  Star-tree）全体としてのエクステント  $EXTx$  の数が少なくなる。この結果当該  $B$ \*木（ $B$  Star-tree）を構成するノード（リーフノード及び中間ノード）の一部が削除される場合がある。従つて



シーケンスマネージャSQMはステップSP5においてB\*木(B Star-tree)を構成するブロック数が減ったか否かを判断する。ここで肯定結果が得られると、このことは減ったブロックに換えて上述のステップSP1においてLRU(Least Recently Used) アルゴリズムによつて選択されなかつた優先度の低いライトキャッシュブロックを含めることができることを表しており、このときシーケンスマネージャSQMは上述のステップSP1に戻つて再びライトキャッシュブロックの選択を繰り返す。

かくしてステップSP1～ステップSP5の処理を不要ブロックがなくなるまで繰り返すことにより、ステップSP5において否定結果を得る。これによりCD-Rディスクの記録領域の利用効率を向上することができる。

ステップSP5において否定結果が得られると、キャッシュマネージャCAMはステップSP6に移つて、ライトキャッシュバッファに集められた複数のブロックからなるパケットデータをデバイスドライバを介してCD-Rドライブ5に送出することにより、CD-RディスクDISCの新たな書込み領域にパケット単位でこれを書き込む。

ここでCD-Rディスクへの書込みエラーが発生したとき、続くステップSP7において肯定結果が得られ、このときシーケンスマネージャSQM及びキャッシュマネージャCAMはステップSP8に移つてCD-Rディスクに対する物理的な修復を行う。すなわちCD-Rディスク装置においては、パケット単位でデータを書込みことが規格で決められており、途中で書込みエラーが発生して記録データが途切れたパケットに対しては、当該パケットを規格の長さまでダミーデータ(RF信号)で埋める作業が必要となる。

従つてシーケンスマネージャSQM及びキャッシュマネージャCAMは不完全なパケットをダミーデータで修復し、当該パケットとして記録されるべきデータを次のパケット記録領域から再び記録し直す。このとき書き込まれるデータの物理アドレスLBAが変更されることにより、シーケンスマネージャSQM及びキャッシュマネージャCAMは上述のステップSP1に戻つてこの時点で未書込み

ライトキャッシュブロックのデータを集め直すとともに新たに集められたライトキャッシュブロックに対して新たな予定物理アドレス(Contact LBA)を割り振る。因みに、エラーが発生する前に書込みが完了したパッケージがあれば、このパッケージに含まれるデータブロックはキャッシュバッファ内において既にリードキャッシュブロックと呼ばれる未更新データブロックに変更されていることにより、ステップS P 8以降のライトキャッシュブロックの集め直しの対象とはならない。従ってパッケージの書込みに成功する度に書込みの候補(ライトキャッシュブロック)が減って行き、最終的に全てのデータブロックがCD-Rディスクに書き込まれることになる。

かくしてステップS P 7において否定結果が得られた(すなわち書込みが成功したパッケージ)に対して、キャッシュマネージャCAMはキャッシュバッファ内の対応するライトキャッシュブロックをリードキャッシュブロックに変更し、続くステップS P 10において全てのパッケージが書き込まれた結果を得るまで上述のステップS P 6～ステップS P 9の処理を繰り返す。

ここで図11はCD-Rディスク上へのデータの記録状態を示し、マルチセッションのパッケージ記録方式においては、図11(A)に示すように、CD-Rディスクの記録エリア上に複数のセッション(Session1、Session2、……)が内周側から外周側に向かって螺旋状に順次記録されて行く。記録エリアの内側には、パワー調整用の情報や各セッションの管理情報を記録し得るようにパワーキャリブレーションエリア(PCA)やプログラムメモリエリア(PMA)が確保されている。

各セッションは、それぞれユーザが作成又は更新したシーケンス(ファイル)のブロックデータ(ファイルデータ)が記録されるプログラムエリア(Program Area)と、セッションの始まりを示すリードイン情報が記録されるリードインエリア(Lead-In Area)と、セッションの終わりを示すリードアウト情報が記録されるリードアウトエリア(Lead-Out Area)とによつて構成されている。なお、リードイン情報及びリードアウト情報は、プログラムエリアにファイルデータ(ユーザデータ)を1つのセッション分だけ記録した後、記録されるものであり、CD-ROM

との互換性を持たせるためのデータである。

図 1 1 (B) に示すように、プログラムエリア内はさらに分かれ、例えば 3 データトラックの場合には 3 つのトラックに分かれる。このとき各トラックの先頭にはインデックスエリア (Index) が設けられ、この部分にトラックのインデックス情報が記録される。また図 1 1 (C) に示すように、トラックはデータ書込みの基本単位であるパケットの集合で構成される。このパケットは図 1 1 (D) に示すように、リンクブロック (Link Block)、ランインブロック (Run In Blocks)、フィアル情報等のユーザデータが記録されるユーザデータブロック (User Data r)、ランアウトブロック (Run Out Blocks) の 4 つの部分に分かれている。

ここで図 1 1 (A) に示すプログラムエリアに記録されるデータとしては、ユーザが作成したシーケンス (ファイル) のブロックデータ (ユーザデータ) の他にデータの管理構造を示す情報が有り、この情報として、プライマリボリュームディスクリプタ (Primary Volume Descriptor: P V D)、スーパーブロック (Super Block)、ノードテーブル (Node Table)、B \* 木インデックスノード (B\*Tree Index Node)、シーケンス B \* 木リーフノード (Sequence B\*Tree Leaf Node)、ディレクトリ B \* 木リーフノード (Directory B\*Tree lleaf Node) がある。なお、ノードテーブルを除く全てのデータ管理構造が 1 ブロック (2048 byte) の大きさを持っており、ブロック境界に記録される。またノードテーブルは 1 ブロック以上の大きさの可変長データ構造であるが、先頭はブロック境界から始まるものである。

プライマリボリュームディスクリプタ P V D はセンシヨンの先頭から 1 6 ブロック目に記録される情報である。その 1 6 ブロック目の先頭 1152 バイトには IS09 660 の P V D と同じものが記録され、1153 バイト目以降には図 1 1 に示すような CDRFS 固有の情報が入っている。図 1 2 の P V D において「Super Block Search Method」は最新のスーパーブロックの格納位置を示すものである。すなわちスーパーブロックは、後述するフラッシュオール動作が行われる度に CD-R ディスク上に書き込まれるようになされており、P V D の「Super Block Search Method

」によつて最新の位置を知ることができる。

例えば「Search Method = 0」である場合には、「Super Block LBA」で示されるブロックに記録されていることを表している。また「Search Method = 1」の場合には、「Last Accessible Block」に記録されていることを表している。また「Search Method = 2」である場合には、「Super Block Area」に記録されているスーパーブロックの中で「Super Block Serial Number」が最大の位置に記録されていることを表している。なお、「Super Block Area」とは「Start LBA of Super Block Area」で示されるブロックと「End LBA of Super Block Area」で示されるブロックとの間に挟まれたブロック全体のことである。

また図12のPVDにおいて示される「File System Flags」が「0x0001」であれば「Addressing Method II」が用いられていることを示し、また「0x0002」であれば「ISO9660 Volume」をCD-RFS から見えないようにしていることを示し、また「0x0003」であればセッションの最初のトラックがTAO (Track At Once) 又は可変長パケットで記録されていることを示している。

また図12のPVDにおいて示される「Packet Size」は固定長パケットのユーザデータのブロック数を示している。但し、このフィールドは「Addressing Method II」が用いられているときにのみ有効となる。

また図12のPVDにおいて示される「Volume Capacity」はフォーマット後にCD-Rディスクに記録できるユーザデータの総ブロック数を示している。なお、この値はオペレーションシステムOSに対してCD-Rディスクの総容量の情報を返す際に利用されるものであり参考値である。

また図13はスーパーブロックの構造を示し、例えば最初の「Super Block Header」には当該ブロックがスーパーブロックであることを示す情報が記録されている。「Search Method = 1」のとき、何らかの不具合で「Super Block」が「Last Accessible Block」に記録されていない場合には、この「Super Block Header」に基づいて古い「Super Block」を探すようになされている。

また「Super Block Flags」はセッションに有効なデータが記録されているか

否かを示している。「Node Table LBA」はノードテーブルが記録されているブロックを示している。ノードテーブルの大きさが2ブロック以上の場合には、当該ノードテーブルは「Node Table LBA」が示すブロックから連続的に記録される。また「Previous Super Block LBA」は前回記録された「Super Block」の位置を示しており、CD-Rディスクの場合、一旦記録されたデータはディスク上から失われることはない。従つて「Previous Super Block LBA」をたどることによつて過去のボリューム状態を知ることができる。

「Sequence B\*Tree Root Node Number」は図5について上述したB\*木(B Star-tree)でなる管理構造(Sequence B\*Tree)のルートノードのノード番号を示す。また「Directry B\*Tree Root Node Number」はファイルマネージャFLMが管理する「Directry B\*Tree」のノードのノード番号を示す。「Serial Number」はスーパーブロックの通し番号である。なお、フォーマット時に生成されるスーパーブロックの「Serial Number」は「0」である。

また「Super Block List」は過去50個の「Previous Super Block LBA」を集めたテーブルであり、図14に示すようなスーパーブロックリストエントリ(Super Block List Entry)の繰り返しで構成される。「Super Block List」の最初のエントリは、「Super Block List」を格納しているスーパーブロックの1つの前のスーパーブロックを示す。これらのエントリはスーパーブロックの通し番号の降順に並んでいる。過去のスーパーブロックが50個に満たない場合は、エントリは先頭から詰められ、未使用のエントリは「0」で埋められる。

「Super Block Tag List」は、過去のスーパーブロックの名称ラベル(タグ)のテーブルであり、図15に示すようなタグエントリ(Tag Entry)の繰り返しで構成される。このCD-RFSでは、1つのCD-Rディスクに最大24個のタグを付けることができ、その際タグが24個に満たない場合には、エントリは先頭から詰められ、未使用のエントリは「0」で埋められる。

またノードテーブル(Node Table)は、図5について上述したように、B\*木(B Star-tree)でなる管理構造(Sequence B\*Tree)の各ノードのノード番号

とその物理アドレス L B A との対応表であり、図 1 6 に示すような構造を有する。このノードテーブルはデータブロックの先頭から連続的に記録され、もし、1 つのデータブロックに納まらない場合には、次のデータブロックに続きが記録される。

またインデックスノードは、図 5 について上述したように B \* 木 (B Star-tree) でなる管理構造のリーフノード以外のノードであり、図 1 7 に示すような情報が記録される。図 1 7 に示される「Number of Records」はリーフノードに格納されているインデックスレコードの数を示している。なお、このインデックスレコードは、図 1 8 に示すような構造を有し、キー(key)の昇順にソートされ、「Index Recoed[0]」から順に詰められて記録される。

またシーケンス B \* 木リーフノードは、図 5 について上述したように、シーケンスキー S Q K と物理アドレス L B A との対応関係を格納するための B \* 木 (B Star-tree) のノードであり、図 1 9 に示すような構造を有する。このリーフノード内のエクステント E X T x は、図 2 0 に示すような構造を有し、キーの昇順にソートされ、「Extent Recoed[0]」から詰めて記録される。なお、ノード内のエクステント E X T x の数は「Number of Records」に記録されている。

またディレクトリ B \* 木リーフノード (Directory B\*Tree Leaf Node) h、ファイル名とシーケンスキー S Q K、ディレクトリ名とディレクトリ番号の対応関係と、ファイルやディレクトリの属性情報とを格納するための B \* 木 (B Star-tree) のノードであり、図 2 1 に示すような構成を有する。このリーフノード内の「Node Number」には、リーフノードの「Node Number」に「0x80000000」を加えたものが格納される。「Number of Records」はこのリーフノードに格納されているディレクトリレコードの数を示している。「Previous Node Number」と「Next Node Number」はそれぞれ一番小さなキー(key)を持つリーフノードと一番大きなキー(key)を持つリーフノードの「Node Number」を示している。仮に該当するノードがなければ「0xffffffff」が記録される。「Total Size of Records」には、後述するインデックスレコードオフセット (Index Records Offset) と

ディレクトリレコードの総バイト数が記録される。

ディレクトリレコードエリア(Directory Records Area)は図 2 2 に示すように使用される。「PosX」はインデックスレコードオフセットと呼ばれ、ディレクトリレコードが格納されている位置をディレクトリレコードエリアの先頭からのバイトオフセットで示している。なお、この「PosX」は 1 バイトである。また「PosX」はそれが示しているディレクトリレコードが持つキー(key)順にソートされ、ディレクトリレコードエリアの先頭から詰めて記録される。

「RecX」はディレクトリレコード本体であり、その位置は特に規定されていない。しかし、CDRFS に採用されているアルゴリズムを用いると、その処理結果として最近作られたディレクトリレコードが最もディレクトリレコードエリアの先頭に近い位置に格納される。「Number of Records」に格納されている値を「Nr」とすると、「PosNr-1」と「End of Free」で示される間の領域は未使用領域であり、ディレクトリレコードの更新や作成の際にはこの領域が用いられる。なお、「PosX」は先頭から、「RecX」は後尾から未使用領域を利用するようになっている。

ディレクトリレコードは、ファイル名とシーケンスキー S Q K、ディレクトリ名とディレクトリ番号の対応関係と、ファイルやディレクトリの属性情報とを格納するための可変長データであり、図 2 3 に示すような構造を有する。なお、このディレクトリレコードは上述したディレクトリレコードエリアに記録される。このディレクトリレコード内のキー(key)はそのディレクトリレコードに割り当てられたものであり、図 2 4 に示すように構成されている。

図 2 4 において、ディレクトリ番号はディレクトリ毎に固有に付けられる数字で、同じディレクトリにある全てのディレクトリレコードは同じディレクトリ番号を持っている。「Hashed Key」はそのディレクトリレコードの名前を次式、

$$P(x) = x^{16} + x^{12} + x^5 + 1 \quad \dots\dots (9)$$

で示す生成多項式で割った余りである。この「Hashed Key」は異なる名前に対して同じ数値になってしまうことがある。これを避けるためCDRFS では、「Sequential Number」を用いている。挿入しようとしたディレクトリレコードと異なる名前にも関わらずディレクトリ番号や「Hashed Key」が等しいディレクトリレコードがB\*木(B Star-tree)内に既に存在した場合、CDRFS はB\*木(B Star-tree)内に既に存在しているディレクトリレコードの「Sequential Number」に対して「1」を加えたものを挿入しようとするディレクトリレコードの「Sequential Number」に設定する。

また図23において、サイズ(Size)はキーとサイズ自信を含むディレクトリレコードのバイト数を示している。また当該図23においてタイプ(Type)はディレクトリレコードの種類を示すためのフィールドであり、図25に示すような5種類がある。またこれとは別に、タイプのビット7が立っていると、そのディレクトリレコードが1つ以上のハードリンクディレクトリレコード(Hard Link Directory Record)から参照されていることを示している。

なお、参考までにファイルディレクトリレコード(File Directory Record)やディレクトリディレクトリレコード(Directory Directory Record)、リンクディレクトリレコード(Link Directory Record)の構造をそれぞれ図26～図28に示す。

かかるB\*木(B Star-tree)でなる管理構造のデータ(管理情報)は、フラッシュオール(Flash All)と呼ばれる書込み動作時においてユーザデータエリア(図11)内にファイルデータ(ユーザデータ)とともに書き込まれる。すなわち、スーパーブロックにはノードテーブルの格納されているブロックの物理アドレスLBAや、B\*木(B Star-tree)でなる管理構造(図5)のルートノード番号等が入っており、管理情報からファイルの内容に至るまでPVDを除くCD-Rディスク上の全てのデータへのリンクはスーパーブロックから始まるようになっている。また図5について上述したように、B\*木(B Star-tree)でなる管理構造のノードを参照するためにはノードテーブルが必要となる。従ってこのよ



うな管理情報を構成するスーパーブロックは、ユーザデータエリアの次回書き込み可能位置の1つ前のパケットの最終ブロックにフラッシュオール動作時に書き込まれる。このフラッシュオールが実行されるタイミングは、予め設定されている所定時間が経過した場合及び所定量以上のデータがCD-Rディスクに書き込まれた場合であり、これによりCD-Rディスク上には所定間隔で管理情報（スーパーブロック）が書き込まれることになる。

図29はフラッシュオールの動作手順を示し、シーケンスマネージャSQM及びキャッシュマネージャCAMはステップSP20から当該処理手順に入り、ステップSP21においてキャッシュバッファ内のライトキャッシュブロックに対して、図10において上述したフラッシュ動作の場合と同様にリストを作成する。すなわちシーケンスマネージャSQMはキャッシュマネージャCAMに対してキャッシュバッファ内のライトキャッシュブロックを全て集めるように要求する。キャッシュマネージャCAMは全ての未書き込みキャッシュブロックと、必要に応じてダミーブロックをリストにした後、キャッシュ管理表を参照して対応するシーケンスキーSQKが昇順になるように整列を行う。このように整列したリストの先頭ブロックから書き込み開始の物理アドレスLBA、書き込み開始の物理アドレスLBA+1、……というように順番に予定物理アドレス(Contact LBA)を割り振る。

シーケンスマネージャSQMは、シーケンスキーSQKと物理アドレスLBAとの対応表に基づいてB\*木(B Star-tree)の更新を行う。ここまでの処理は、B\*木(B Star-tree)の更新によつてブロックの削除が生じなくなるまで繰り返えられる。次にステップSP22において、シーケンスマネージャSQMがキャッシュマネージャCAMに対してノードテーブル及びスーパーブロックを格納するためのデータブロックの生成を通常のブロック生成と同じ手続きで要求する。これによりキャッシュマネージャCAMはキャッシュバッファ内にスーパーブロック及びノードテーブル用のブロックを生成する。

ここでノードテーブルのブロック生成には、シーケンスキーSQKとして「ff

ffffff00000000(hex)」、「ffffffff00000001(hex)」、……が渡され、スーパーブロックのブロック生成には、シーケンスキーS Q Kとして「ffffffffffffffffff(hex)」が渡される。このようにシーケンスキーS Q Kを付けることにより、キャツシユマネージャC A Mのブロック整列動作の際に、ノードテーブルが連続領域に、スーパーブロックが最後のパケットの最終ブロックに配置される。

続いてキャツシユマネージャC A MはステップS P 2 3において、キャツシユバッファ内のライトキャツシユブロックについて再度リストを作成する。すなわちシーケンスマネージャS Q MがキャツシユマネージャC A Mに対してキャツシユバッファ内のライトキャツシユブロックを全て集めるように要求する。もし、先に書込みを見送られたデータブロックがあれば、ここで改めて集められることになる。そして通常通りにキャツシユマネージャC A Mによるブロックリストの作成と物理アドレスL B Aの決定、シーケンスマネージャS Q MによるB \* 木(B Star-tree)の更新という一連の処理が行われた後、ステップS P 2 4においてスーパーブロックとノードテーブル用のブロックに各内容を埋め込む。

この後は、ステップS P 2 5以降に進んで実際にデータをパケット単位で記録して行き、全て書き終えたらステップS P 3 0に進んでフラッシュオール動作を終了する。因みにステップS P 2 5～ステップS P 2 9は、それぞれフラッシュ動作のステップS P 6～ステップS P 1 0と同様であるため、説明を省略する。

このようにフラッシュオール動作においてスーパーブロックに対してはシーケンスキーS Q Kとして「ffffffffffffffff(hex)」のように最も大きな値を割り当てるとともに、ノードテーブルに対してはシーケンスキーS Q Kとして「ffffff00000000(hex)」、「ffffffff00000001(hex)」、……のように大きな値でかつ連続した値を割り当て、さらにリーフノードに対してはシーケンスキーS Q Kとして「fffffffe00000000(hex)」、「fffffffe00000001(hex)」、……のようにノードテーブルとは明確に異なる大きな値で連続した値を割り当てる。

これらの大きな値は、スーパーブロック、ノードテーブル及びリーフノード以外のシーケンス(ファイル)を構成するブロックデータ(ユーザデータ)に割り

当てられるシーケンスキー S Q K (「0000000500000000(hex)」、「0000000500000002(hex)」、……) に比べて格段的に大きな値である。従つてこのようにシーケンスキー S Q K を割り当てた状態でフラッシュオール動作を行うと、ユーザデータと管理情報(スーパーブロック、ノードテーブル及びリーフノード)とがシーケンスキー S Q K が昇順に並ぶようにブロックソート(並び換え)されて CD-R ディスク上にソート順に順次記録されるため、大きなシーケンスキー S Q K の値が割り当てられたスーパーブロック、ノードテーブル及びリーフノード等の管理情報は CD-R ディスク上にパケット単位で記録される際、後方のパケットに連続して書き込まれる。特に最も大きなシーケンスキー S Q K が割り当てられたスーパーブロックは、最後のパケットの最終ブロックに書き込まれる。

このように CDRFS では、CD-R ディスク上に後から書き込まれたブロックから先に書き込まれたブロックに対する方向にのみ参照関係を成立させている。従つて例えば図 30 (A) に示すように、左(CD-R ディスクの内周側)から右(CD-R ディスクの外周側)に向かつてデータをパケット単位で順次書き込んでいるとき、これから新たな未書込み領域に書き込もうとしているパケットデータ b (例えばスーパーブロック) が物理アドレス上前方に書き込まれたパケットデータ a (例えばノードテーブル) を参照するように参照関係が一方向のみに成り立っていることにより、図 30 (B) に示すようにパケットデータ b の書き込みが失敗した場合、図 30 (C) に示すように書き込みに失敗したパケットデータ b のみを新たな未書込み領域に書き直すだけで良い。これにより、前方のパケットデータ a が後方のパケットデータ b を参照するようになされている場合に、後方のパケットデータ b の書込み位置を変更することに応じて前方のパケットデータ a 内の後方パケットデータ b の物理アドレス情報を書き換える(CD-R の場合前方パケットデータ a も後方パケットデータ b と共に新たな未書込み領域に書き込む) 必要が有ることに比べて、エラー発生時の書き換え処理を簡単にし得る。

また CDRFS では、CD-R ディスクを CD-R ドライバ 5 から取り出す際に必ずフラッシュオール動作を行つて、最後のパケットにスーパーブロックを書き込むように

なされている。従つて当該CD-Rディスクを再びCD-Rドライブ5に挿入したとき、CDRFSは当該CD-Rディスクをデータが記録された最も外周側の領域からスーパーブロックの探索を行い、図5について上述した当該スーパーブロックの管理情報に基づいて当該スーパーブロックの前方（内周側）に書かれているデータを参照する。このとき、最終パケットにスーパーブロックが書き込み不良を起こしているとき、CDRFSは当該スーパーブロックから順次内周側に向かって1つ前のスーパーブロックの探索を行う。この場合、1つ前のスーパーブロック以降に書かれているユーザデータをアクセスすることが困難になる。従つて本発明のCDRFSでは、CD-Rディスクに対してデータを書き込む際のフラッシュオール動作を所定時間が経過したとき及び、所定データ量以上のデータ量がキャッシュマネージャCAMからCD-Rディスク上に書き込まれたときに行うようにしていることにより、CD-Rディスク上に比較的頻繁にスーパーブロックを書き込まれ、これにより最後のパケットに存在するはずのスーパーブロックが書き込み不良を起こしている場合でも、当該スーパーブロックに対して比較的近い位置に書き込まれている1つ前のスーパーブロックを読み出すことにより、大量のユーザデータが失われることを回避し得る。

ここで、図11について上述したように、CDRFSフォーマットでデータが書かれたCD-RディスクをCD-ROMドライブで読む場合、各セッションの前後にリードイン情報及びリードアウト情報を書き込んでおく必要がある。この動作をフリーズ(Freeze)と呼ぶ。このフリーズ動作を行うためにはCD-Rディスク上に所定の空き容量が必要となり、この容量が確保できない場合にはリードアウト情報を書き込むことが困難となり、当該CD-RディスクをCD-ROMドライブで読む場合、前回リードアウトが確実に記録されているセッションの内容までしか読み出すことができないことになる。

従つてCDRFSではCD-Rディスクの未使用領域の容量監視を行う。すなわちキャッシュバッファ内の未書き込みデータブロックの数をNd、ノードテーブルのブロック数をNnとすると、フラッシュオール動作に必要な空き容量Nfaは、スーパーブ

ロックのためのブロックも考慮して、次式

$$Nfa = \{(Nd + Nn + 1 + 31)/32\} \times (32 + 7) \quad \dots\dots (10)$$

となる。

フリーズに関してはセッションへの書込みを開始した時点でリードイン情報のための領域は確保されることから、ここではリードアウト情報のための容量についてのみ監視すれば良い。CD-ROMの規格からフリーズに必要な空き容量は、セッションが1つであれば、 $1.5 \times 60 \times 75 = 6750$  block、2つ目以降であれば $30 \times 75 = 2250$  blockである。従ってCD-Rディスク上に確保されるべき空き容量の最小値Nfree は、次式、

$$Nfree = Nfa + 6750 \quad (1st \text{ Session}) \quad \dots\dots (11)$$

$$Nfree = Nfa + 2250 \quad (Nth \text{ Session, } Nth \geq 2nd) \quad \dots\dots (12)$$

となる。因みにこの数字は書込みエラーが発生することを考慮しておらず、例えばnパケットの書込みに対して書込みエラーがe回発生すると考えると、書込みエラーの確率Peは、次式、

$$Pe = n / e \quad \dots\dots (13)$$

となる。ここで1回の書込みエラーにつき1つのパケットが余分に消費されることから、書き込みエラーを考慮した空き容量Nfa について補正が加えられ、(11)式及び(12)式は、次式、

$$Nfree = Nfa(1 + Pe) + 6750 \quad (1st \text{ Session}) \quad \dots\dots (14)$$

$$N_{\text{free}} = N_{\text{fa}}(1 + P_e) + 2250 \quad (\text{Nth Session, } N_{\text{th}} \geq 2 \text{ nd}) \quad \dots\dots (15)$$

となる。

かくしてシーケンスブロックの作成、更新又は消去動作の後で、 $N_{\text{free}}$  ブロック数以上の空き容量がCD-Rディスク上に確保されていれば、フリーズ動作を行うことができる。

CDRFS では、シーケンスマネージャSQMに対する1回の呼び出しで新たに生成されるキヤツシユバッファ内のライトキヤツシユブロック数の最大値は(6)式について上述したように  $MBC_{\text{max}}(N_{\text{max}})$  であるから、空き容量が  $N_{\text{free}} + MBC_{\text{max}}(N_{\text{max}})$  ブロック数以下のCD-Rディスク等の記録メディアがドライバに挿入されている場合には、ユーザに対して記録メディアの容量不足を警告し、シーケンスブロックの作成、更新又は消去動作等の記録メディア(CD-Rディスク等)に対する書込みを禁止する。これによりCDRFS はフリーズ動作を保証することができる。

但し、CDRFS が提供するファイルアクセスのためのサービスは、シーケンスブロックへの複数回の操作から構成されており、任意のシーケンスブロックに対する操作終了時に記録メディア(CD-Rディスク)に対しての書き込み禁止を行ってしまうと、ファイアルシステムとしての整合性が失われてしまうおそれがある。従って記録メディア(CD-Rディスク)に対する書き込み禁止動作は、オペレーティングシステムOSからCDRFS に対しての動作要求の切れ目で行う。このときオペレーティングシステムOSからの1回のCDRFS に対する動作要求によつて実行されるシーケンスブロックの作成、更新又は消去作業の回数を  $N_{\text{cb}}$  とすると、記録メディア(CD-Rディスク)に対する書き込み禁止の空き容量の条件は、次式、

$$N_{\text{free}} = MBC_{\text{max}}(N_{\text{max}}) \times N_{\text{cb}} \quad \dots\dots (16)$$

で表される値Nfree 以下となる。

以上の監視方法では、CD-ROMとの互換性を持たせるためのリードアウト情報を最後にCD-Rディスク上に書き込むための容量を確保しているが、当該リードアウト用の容量に加えてダテイブロック分を含めて監視しても良い。すなわちかかる監視動作は、図31に示す監視処理手順に従ってCD-Rディスク上にデータが書き込まれる度に行われる。すなわちCDRFS は図31のステップSP40から当該監視処理手順に入り、ステップSP41においてCD-Rディスクの残り容量を確認する。すなわちCD-Rディスク上には当該CD-Rディスクの製造過程で全体の記録可能容量が記録されている。従ってCD-Rドライバ5は当該CD-Rディスクから全体の記録可能容量と、現在記録されている最終物理位置から記録可能残容量を求める。当該求められた記録可能残容量は、ステップSP42においてキャツシユマネージャCAMに渡され、キャツシユマネージャCAMはキャツシユバッファ内の書き込み予定データ量（ダテイブロック）を記録可能残容量から差し引き、その残り容量をシーケンスマネージャSQMに渡す。シーケンスマネージャSQMはステップSP43において、B\*木（B Star-tree）管理構造のスーパーブロック（SVD）及びノードテーブルの書き込み量を、キャツシユマネージャCAMから渡された残り容量から差し引き、その残り容量をファイルマネージャFLMに渡す。ファイルマネージャFLMはステップSP44において、リードアウト情報等を記録させるために必要となるマージン分を、シーケンスマネージャSQMから渡された残り容量から差し引く。

このようにして算出された残り容量が「0」よりも大きい場合、CDRFS は書き込み処理が可能であると判断し、ステップSP46に移って処理を続行する。これに対してステップSP45において否定結果が得られると、このことは、CD-Rディスク上にこれから処理しようとするデータを書き込むだけの余裕がないことを表しており、このときCDRFS はステップSP47に移ってユーザに対して所定の警告を表示する。かくしてCDRFS はCD-Rディスク上にフラッシュ動作又はフラッシュオール動作が実行される度に、実行後のディスク上の残り容量を確認し、

その後の処理での書き込み量に対して余裕がなくなると判断される場合にはユーザに対して警告を発することにより、書き込むために余裕のないCD-RディスクをCD-Rドライブ5に挿入したままでその後の処理が実行されることを未然に防止し得る。

またCD-Rディスクを用いるCDRFSでは、CD-Rディスクをあたかも書き換え可能なメディアのように見せかけるが、CD-Rディスクの場合、更新作業のたびに新たな領域を使用することにより、当該CD-Rディスクの利用可能残容量が減ることになる。従って当該利用可能残容量は、新たなデータが書き込まれる度に監視される。この残容量は、キヤツシュバツファ内のダーティブロックの他にB\*木(B Star-tree)による管理構造の要素であるスーパーブロック及びノードテーブルと、CD-ROMとの互換性を持たせるためのリードアウト情報とを含むものであり、これらの情報を記録し得る残容量を監視することによつて、ユーザが仮想アドレス空間上で作成又は更新中のデータを失わせることなく、確実にCD-Rディスク上にCD-ROMとの互換性を持たせた状態で記録することが保証される。

以上の構成において、CDRFSはCD-Rディスク上の物理アドレスLBAを仮想アドレス空間において論理アドレスであるシーケンスキーSQKと対応させて管理する。従って例えば更新が指示されたCD-Rディスク上のブロックデータは、キヤツシュバツファ内に読み出されてデータが更新された後、そのシーケンスキーSQKは変更されずに物理アドレスLBAだけが変更され、所定のタイミング（フラッシュ又はフラッシュオール動作時）においてCD-Rディスク上の新たな物理アドレスLBA位置に書き込まれる。

このときファイルマネージャFLMは仮想アドレス空間上で論理アドレスを用いてデータの更新、作成、消去等のデータ処理を行うことになり、この結果、使用するメディアがライトワンス型のCD-Rディスクであつても、これを認識することなく当該CD-Rディスクがあたかも従来の書き換え可能なメディアであるかのように処理を行うことができる。かくしてファイルマネージャFLMの処理機能を、記録媒体に左右されることのないモジュールにすることができ、ライトワンス



型メディアの特殊な記録処理に煩うことなく、ディレクトリやファイル管理のみを確実に行うことができる。

また図3について上述したように、複数の仮想アドレス空間CV1、CV2、CV3をそれぞれファイル（シーケンス）に割り当て、1つの仮想アドレス空間を1つのファイルに対応させる。これにより仮想アドレス空間がファイルそのものとなることにより、ファイルを論理アドレスに変換する作業が不要となる。因みに図32は、仮想アドレス空間AVをシーケンス処理ブロックC2によつて記録媒体上の物理的地址位置ARに変換する場合を示し、この方法においては仮想アドレス空間がファイルごとに分けられておらず、図3の場合に比べてファイルを論理アドレスに変換する作業が必要になる。従つて本発明によれば図3のようにファイルごとに仮想アドレス空間を割り振ることにより、書き込み可能メディア等に比べてファイルへのアクセス性能を低下させることなく、ライトワンス型に対応するファイルシステム（CDRFS）を構築することができた。

以上の構成によれば、CD-Rディスク上の新たな物理アドレスLBAに書き込まれた更新データブロックをアドレスの交代処理機能を有するCDRFSを用いてあたかも更新前のアドレスにあるデータブロックのように見せることにより、ライトワンス型記録メディアであるCD-Rディスクを疑似的に書き換え可能な従来の記録メディアとして扱うことができる。

また、後から書き込むパッケージが先に書き込まれたパッケージを参照するように参照関係を一方向に制限したことにより、書き込みが失敗したとしても他のパッケージに影響を及ぼさずに書き直しをすることができる。

またキヤツシュバツファ内のダーティブロックを集めてパッケージを生成する場合、LRUアルゴリズムに基づく優先度に従つてダーティブロックを集め、集められたダーティブロックに対して各シーケンスキーSQKが昇順となるように並べて予定物理アドレス（Contact LBA）を割り振ることにより、当該ダーティブロックがCD-Rディスク上に書き込まれたとき、連続領域にシーケンスが並ぶ確率が増え、読出し性能の向上を図ることができる。

またCD-Rディスクの利用可能残容量を監視することにより、ユーザが作成中のデータは確実にCD-Rディスク上に記録する保証が得られる。

なお上述の実施例においては、ライトワンス型の記録メディアとしてCD-Rディスクを用いた場合について述べたが、本発明はこれに限らず、他のライトワンス型の記録媒体であつても良い。

また上述の実施例においては、CD-Rディスクの利用可能残容量を監視する際に当該残容量に余裕がなくなつたとき警告を発するようにしたが、本発明はこれに限らず、当該警告を発するタイミングでフラッシュオール動作及びフリーズ動作を行つて処理済のデータをCD-Rディスク上に確実に記録するようにしても良い。

また上述の実施例においては、CDRFS によつて1つのCD-Rディスクに対してデータの書き込みを実行する場合について述べたが、本発明はこれに限らず、例えば複数の記録メディアを設定してマルチデバイス化し、1つのCDRFS によつて複数のメディアを管理するようにしても良い。これにより例えば複数のメディアを管理するチェンジャや複数のドライブを管理するジューク等を1つのファイルシステムによつて容易に管理することができる。

#### 産業上の利用可能性

本発明は、情報を処理して記録媒体に記録するシステムにおいて、追記型記録媒体を用いた情報処理装置に利用できる。

## 請 求 の 範 囲

1. 媒体上に記録されるファイルデータの物理位置を仮想空間上の論理位置と対応させるとともに、上記物理位置及び上記論理位置の対応表を生成し、上記物理位置が変更されたとき上記論理位置と変更後の物理位置との対応関係をメモリ上で管理する管理手段と、

上記メモリ内のファイルデータ及び上記対応表の要素データを書込みデータとして上記媒体に記録する記録手段と、

上記媒体の書込み可能残容量を算出する算出手段と、

上記書込み可能残容量と上記メモリ内の上記書込みデータ量とを比較する比較手段と、

上記比較手段の比較結果が書込み量の超過を示すとき、当該超過が検出された結果を超過検出結果として出力する出力手段と

を具えることを特徴とする情報処理装置。

2. 上記記録手段は、

上記超過検出結果が出力されたとき、直ちに上記メモリ内の上記書込みデータを上記記録媒体上に記録する

ことを特徴とする請求の範囲第1項に記載の情報処理装置。

3. 上記情報処理装置は、

上記超過検出結果が出力されたとき、所定の表示手段に警告表示する警告手段を具える

ことを特徴とする請求の範囲第1項に記載の情報処理装置。

4. 上記媒体は、追記型記録媒体であり、

上記書込みデータは、書換え可能型記録媒体に必要なフォーマット情報を含むことを特徴とする請求の範囲第1項に記載の情報処理装置。

5. 上記管理手段は、

上記ファイルデータの論理位置を示すキー情報と、上記キー情報で指示される

ファイルデータの上記媒体上での物理位置との対応関係を示す対応情報を複数収納するファイル情報テーブルを複数生成するファイル情報テーブル生成手段と、

上記複数のファイル情報テーブルのうち、上記各キー情報をそれぞれ収納したファイル情報テーブルを示すインデックステーブルを生成するインデックステーブル生成手段と、

上記複数のファイル情報テーブル及び上記インデックステーブルの上記媒体上での物理的記録位置を示す管理テーブルを生成する管理テーブル生成手段と

を具え、上記ファイル情報テーブルと上記インデックステーブルと上記管理テーブルとを上記対応表の要素データとして上記メモリ上で管理する

ことを特徴とする請求の範囲第1項に記載の情報処理装置。

6. 上記キー情報は、

上記ファイルデータのそれぞれに固有に付けられたシーケンス番号及びブロック番号である

ことを特徴とする請求の範囲第5項に記載の情報処理装置。

7. 媒体上に記録されるファイルデータの物理位置を仮想空間上の論理位置と対応させるとともに、上記物理位置及び上記論理位置の対応表を生成し、上記物理位置が変更されたとき上記論理位置と変更後の物理位置との対応関係をメモリ上で管理する管理工程と、

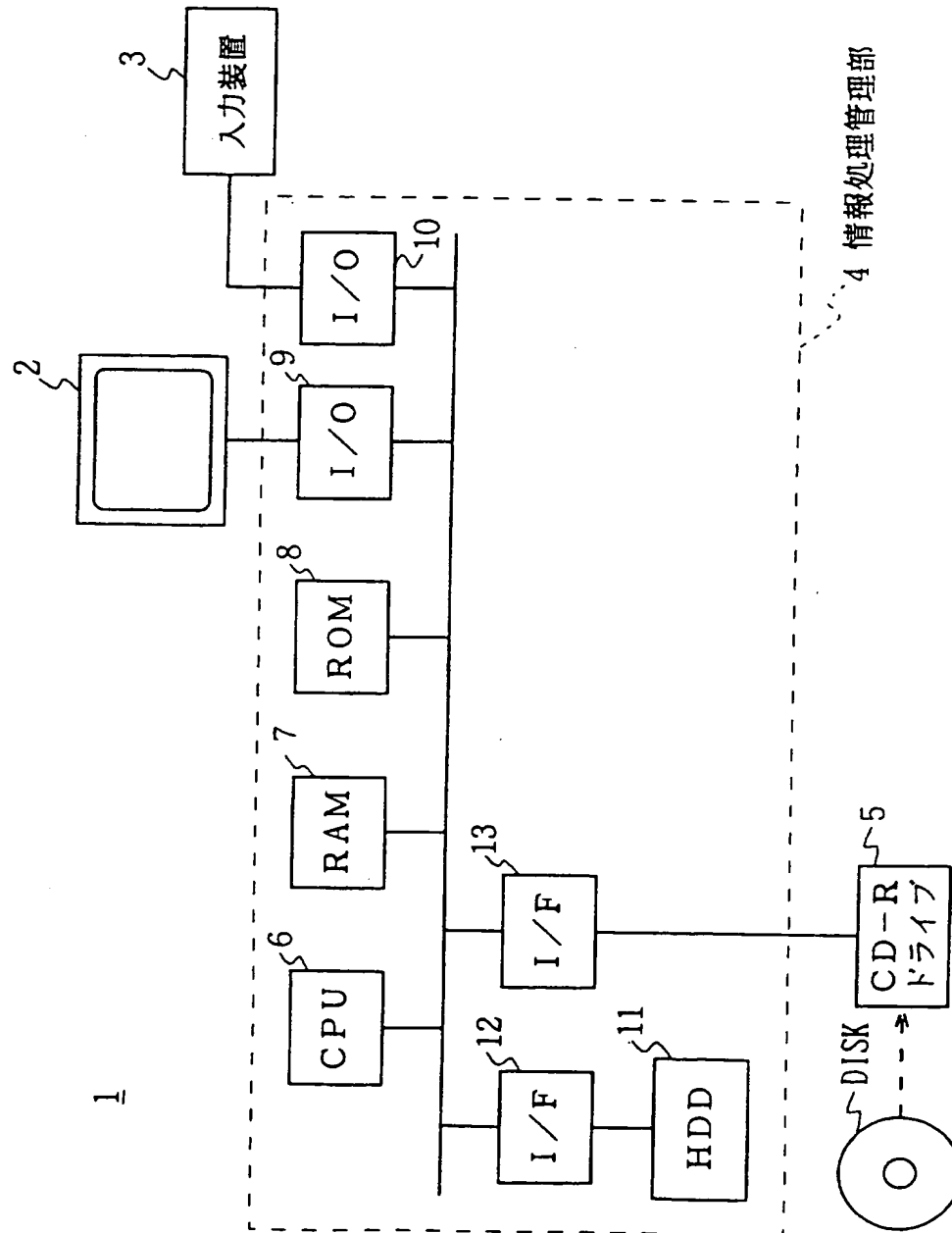
上記メモリ内のファイルデータ及び上記対応表の要素データを書込みデータとして上記媒体に記録する記録工程と、

上記媒体の書込み可能残容量を算出する算出工程と、

上記書込み可能残容量と上記メモリ内の上記書込みデータ量とを比較する比較工程と、

上記比較手段の比較結果が書込み量の超過を示すとき、当該超過が検出された結果を超過検出結果として出力する出力工程と

を具えることを特徴とする情報処理方法。



1  
☒

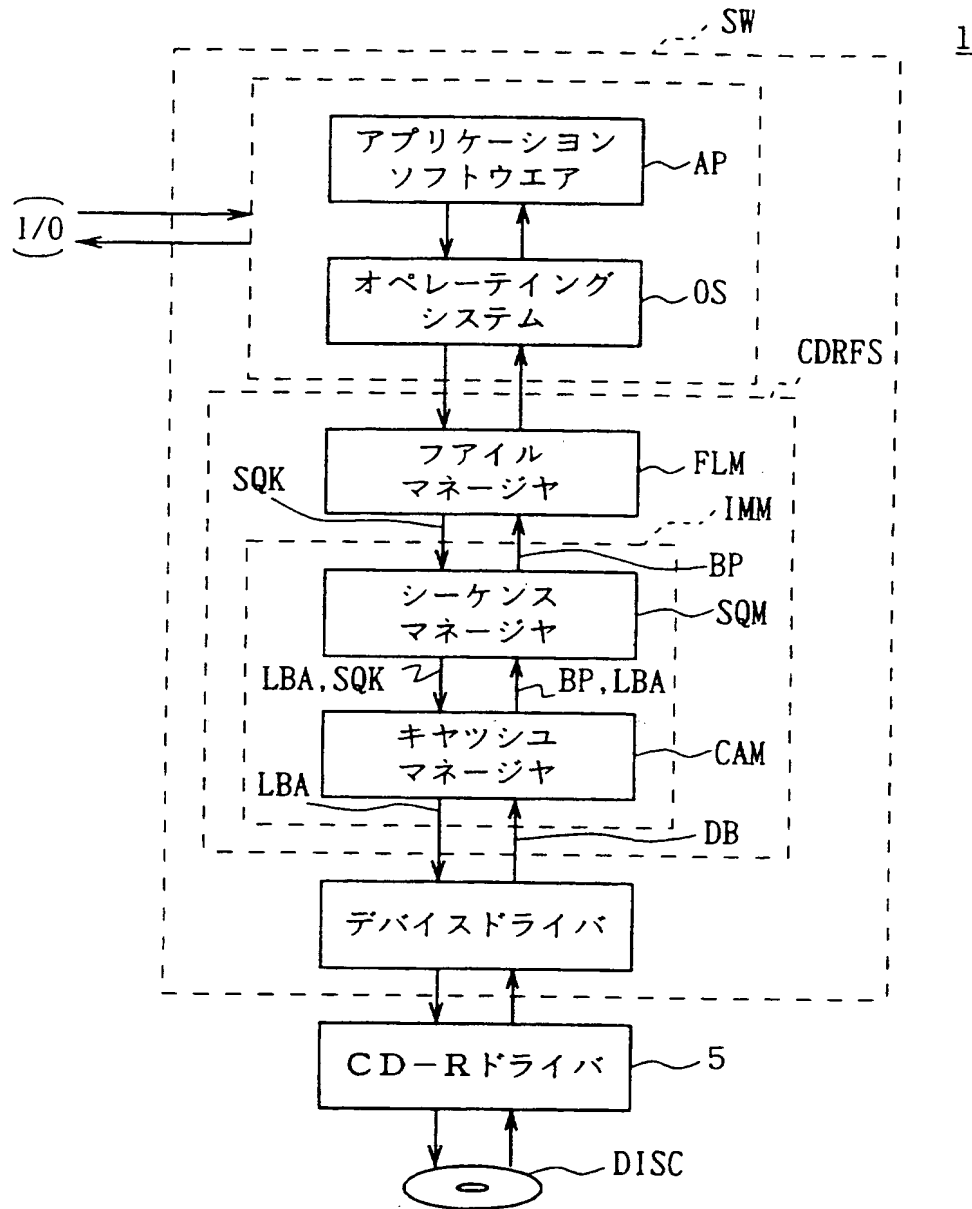


図 2

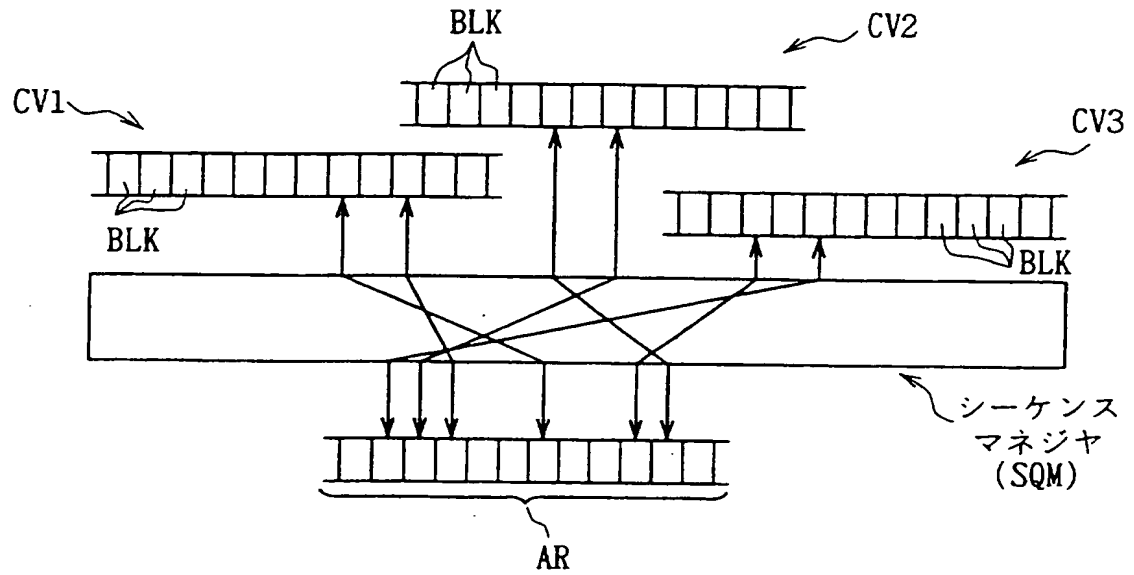


図 3

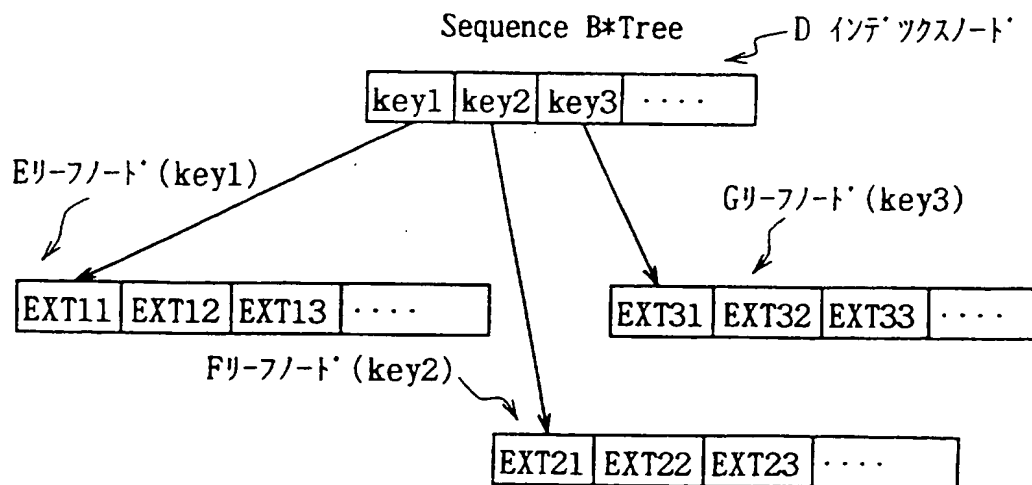
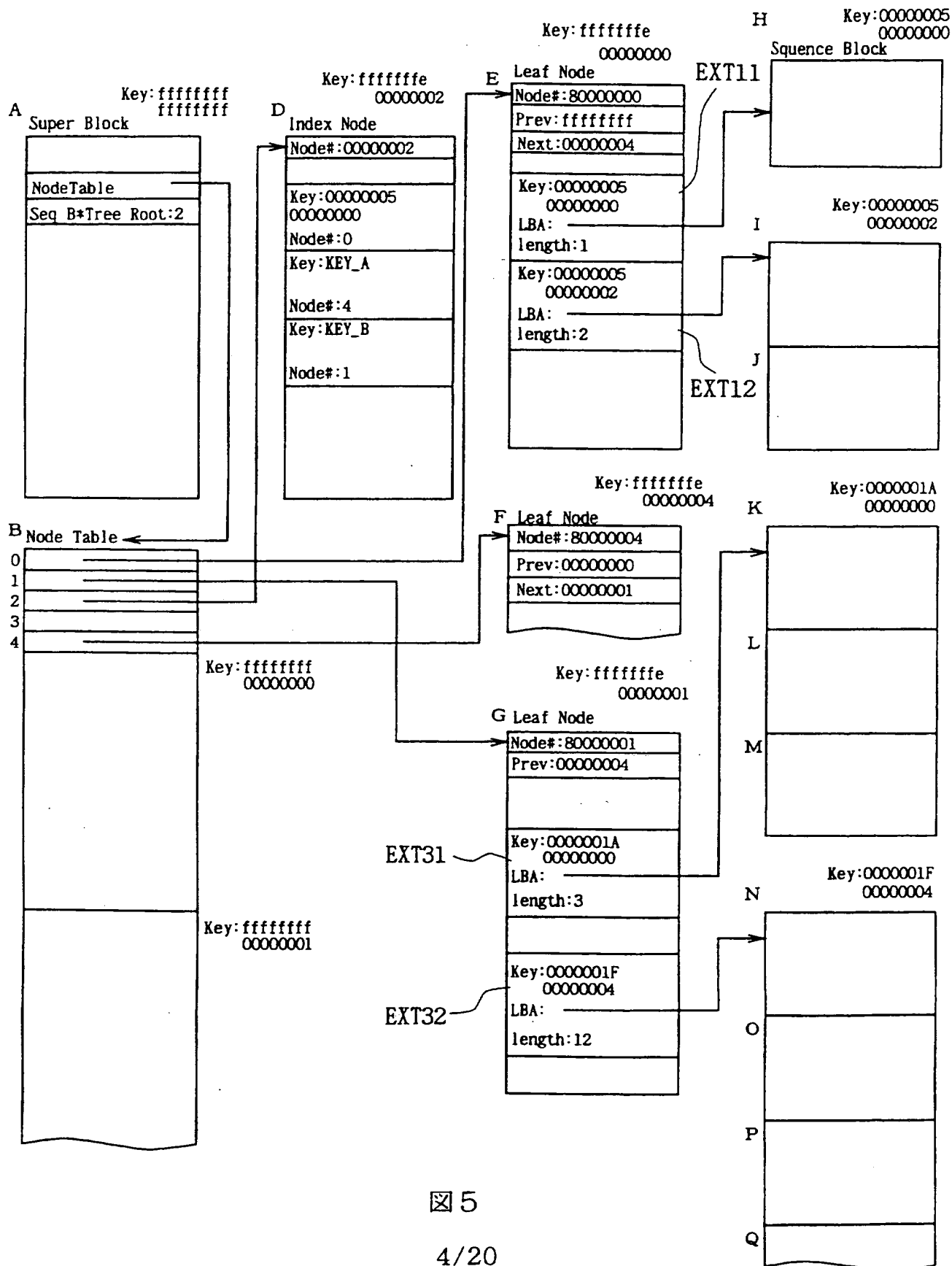


図 4





Sequence Number		LBA
123456789abcdef0h	→	1000h
123456789abcdef1h	→	1001h
123456789abcdef2h	→	1002h
123456789abcdef3h	→	1003h
123456789abcdef4h	→	1004h

図 6

number of entry	6
free	2
table[0]	30
table[1]	6
table[2]	4
table[3]	4
table[4]	1
table[5]	22

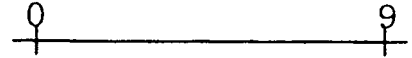
図 7

Leaf Node	$1 \times 2^{20} \text{ kbyte} / 2 \text{ kbyte} / (145 \text{ extent} / 2)$ = 7231.6 block = 7232 block
Index Node (3rd level)	72312 block / (170 index / 2) = 850.7 block = 851 block
Index Node (2nd level)	851 block / (170 index / 2) = 10.01 block = 11 block
Index Node (root)	1 block
total	8095 block

図 8

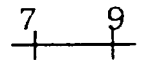
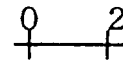
☒ 9 ( A )

Sequence Key	LBA	Length
0	3	10



☒ 9 ( B )

Sequence Key	LBA	Length
0	3	3
7	10	3



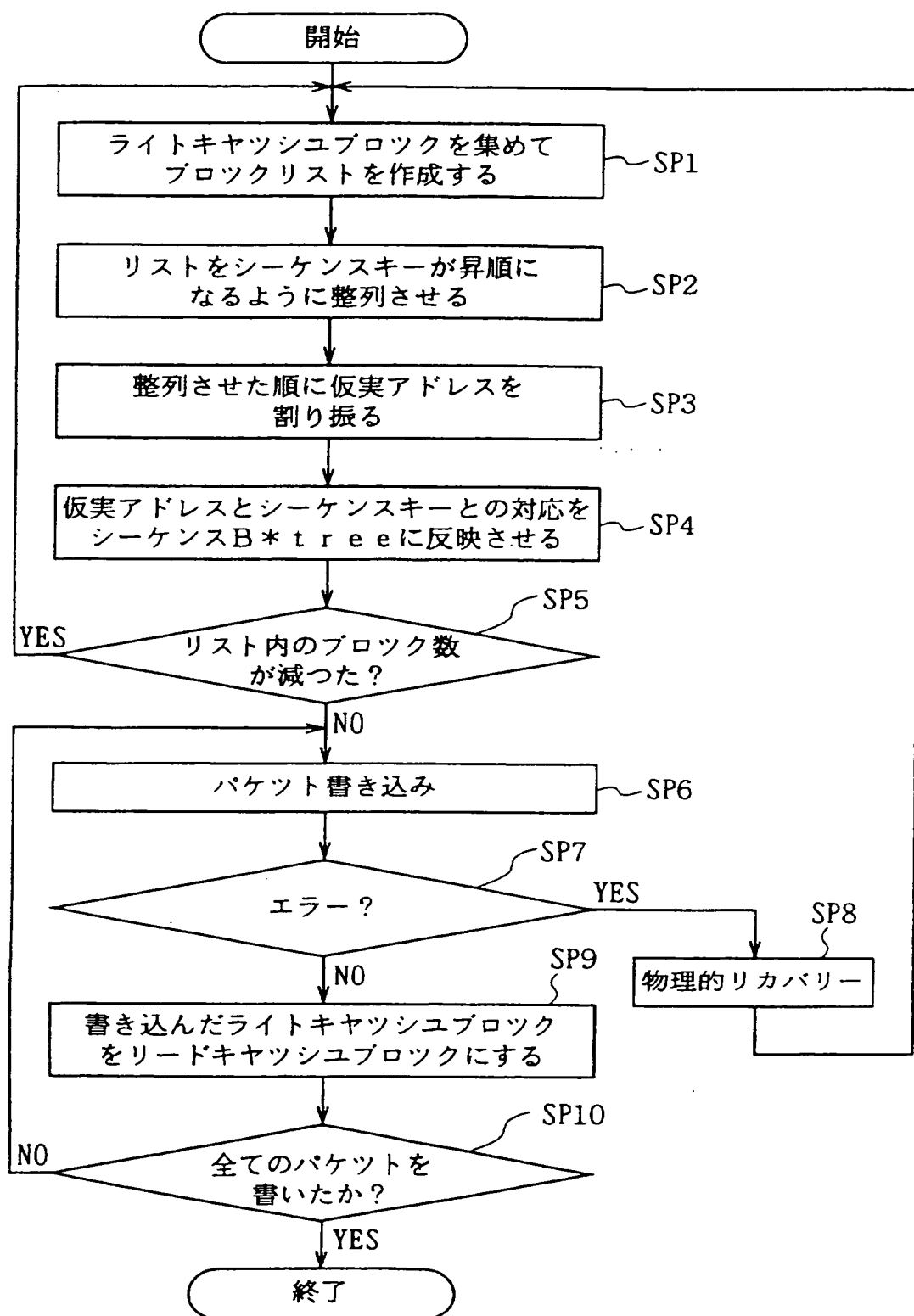
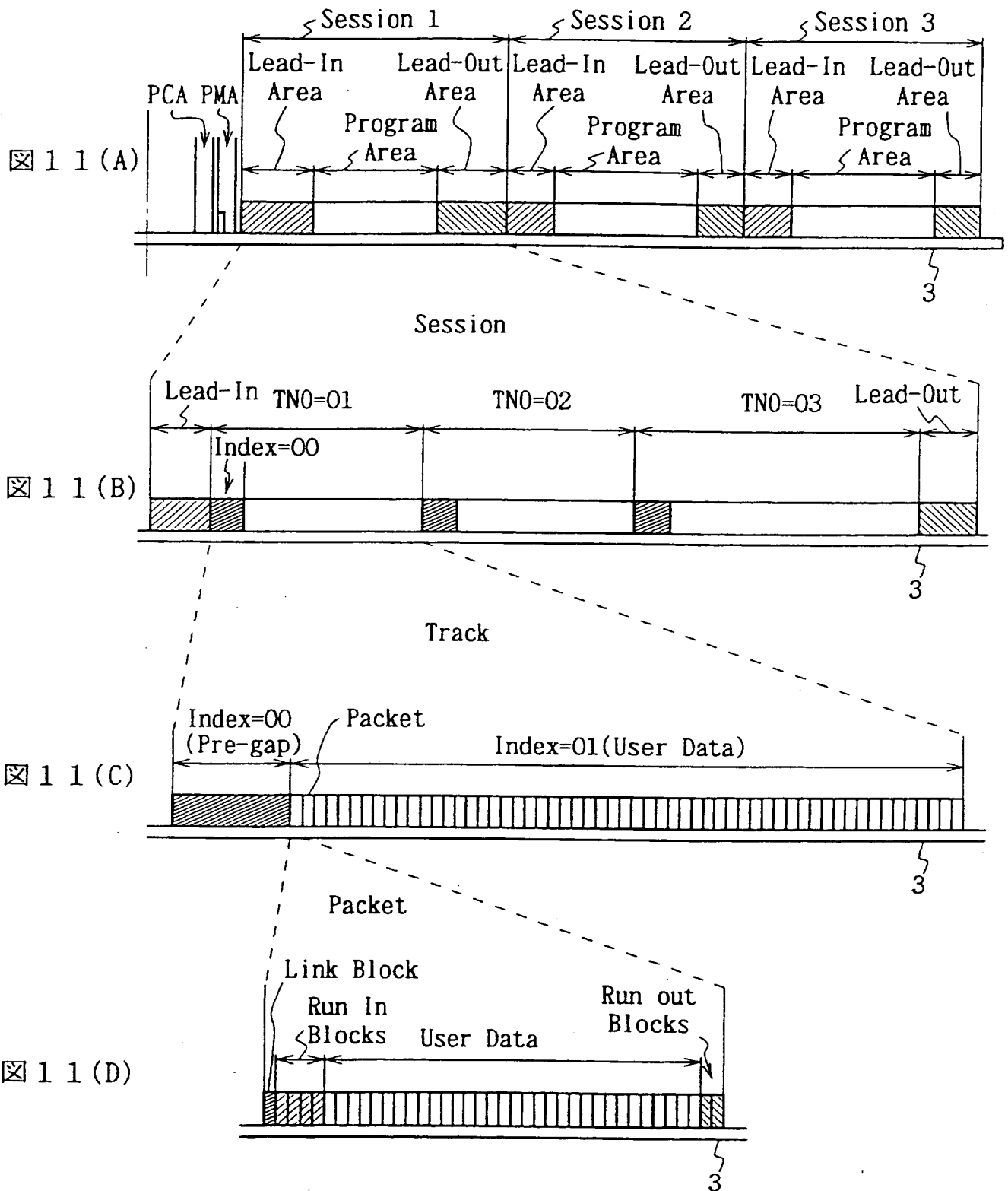


図 10



Start	Length	contents
1152	5	File System Standard ID (= "CDRFS")
1157	1	File System Version
1158	16	Copyright (= "Sxxx CORPORATION")
1174	1	Super Block Search Method
1175	1	Reserved (= 0)
1176	4	Super Block LBA
1180	4	Start LBA of Super Block Area
1184	4	End LBA of Super Block Area
1188	2	File System Flags
1190	2	Packet Size (Addressing Method-II only)
1192	4	Volume Capacity
1196	852	Reserved (= 0)

図 1 2

Start	Length	contents
0	64	Super Block Header
64	2	Super Block Version
66	2	Super Block Flags
68	4	Node Table LBA
72	4	Previous Super Block LBA
76	4	Sequence B*Tree Root Node Number
80	4	Directory B*Tree Root Node Number
84	4	Serial Number
88	400	Super Block List
488	1536	Super Block Tag List
2024	20	Reserved (= 0)
2044	4	Check Sum

図 1 3

Start	Length	contents
0	4	Super Block LBA
4	4	Creation Time

図 1 4

Start	Length	contents
0	4	Super Block LBA
4	4	Creation Time
8	56	Tag String

図 1 5

Start	Length	contents
0	4	Number of Entry (= Ne)
4	4	Free Entry
8	4	entry[0]
12	4	entry[1]
16	4	entry[2]
•	•	•
•	•	•
•	•	•
4Ne+8	4	entry[Ne-1]

図 1 6

Start	Length	contents
0	4	Node Number
4	1	Number of records
5	3	reserve (0)
8	12	Index Record[0]
20	12	Index Record[1]
•	•	•
•	•	•
•	•	•
2036	12	Index Record[169]

図 17

Start	Length	contents
0	8	Key
8	4	Node Number

図 18

Start	Length	contents
0	4	Node Number
4	1	Number of records
5	3	reserved (= 0)
8	4	Previous Node Number
12	4	Next Node Number
16	2	reserved (= 0)
18	14	Extent Record[0]
32	14	Extent Record[1]
•	•	•
•	•	•
•	•	•
2034	14	Extent Record[144]

図 19

Start	Length	contents
0	8	Key
8	4	LBA
12	2	Length

図 20



Start	Length	contents
0	4	Node Number
4	1	Number of records
5	1	reserve (0)
6	2	Total Size of Records
8	4	Previous Node Number
12	4	Next Node Number
16	2	End of Free Area
18	2030	Directory Record Area

図 2 1

PosX:Directory Record Offset

RecX:Directory Record

Free:Free Area

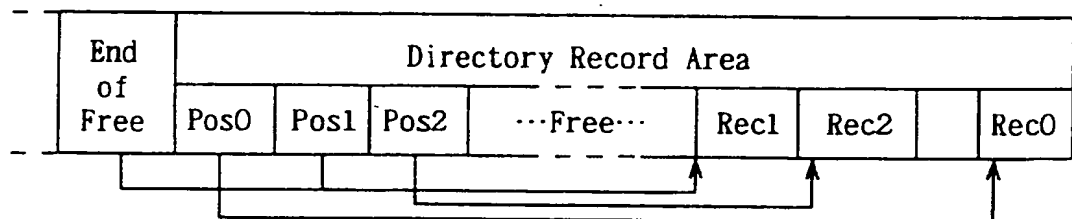


図 2 2

Start	Length	contents
0	8	Key
8	2	Size
10	1	Type
11	1	Attribute
12	2	Mode
14	Size - 14	Other Attribute

図 2 3

Start	Length	contents
0	4	Directory Number
4	2	Hashed Key
6	2	Sequential Number

図 2 4

Type	種類
Type = 0	File
Type = 1	Directory
Type = 2	Volume Label (Root Directory)
Type = 3	Symbolic Linc
Type = 4	Hard Linc

図 2 5

Start	Length	contents
0	8	Key
8	2	Size
10	1	Type
11	1	Attribute
12	2	Mode
14	2	Number of Links
16	2	User Name
18	2	Group Name
20	4	Last Access Time
24	4	Last Modification Time
28	4	Creation Time
32	8	File Size
40	4	Sequence Number
44	1	Name Length (=NL)
45	NL	File Name

図 2 6

Start	Length	contents
0	8	Key
8	2	Size
10	1	Type
11	1	Attribute
12	2	Mode
14	2	Number of Links
16	2	User Name
18	2	Group Name
20	4	Last Access Time
24	4	Last Modification Time
28	4	Creation Time
32	4	DirectoryID
36	1	Name Length (=NL)
37	NL	Directory Name

図 27

Start	Length	contents
0	8	Key
8	2	Size
10	1	Type
11	1	Reserve (=0)
12	8	Link
20	1	Name Length (=NL)
21	NL	Link Name

図 28

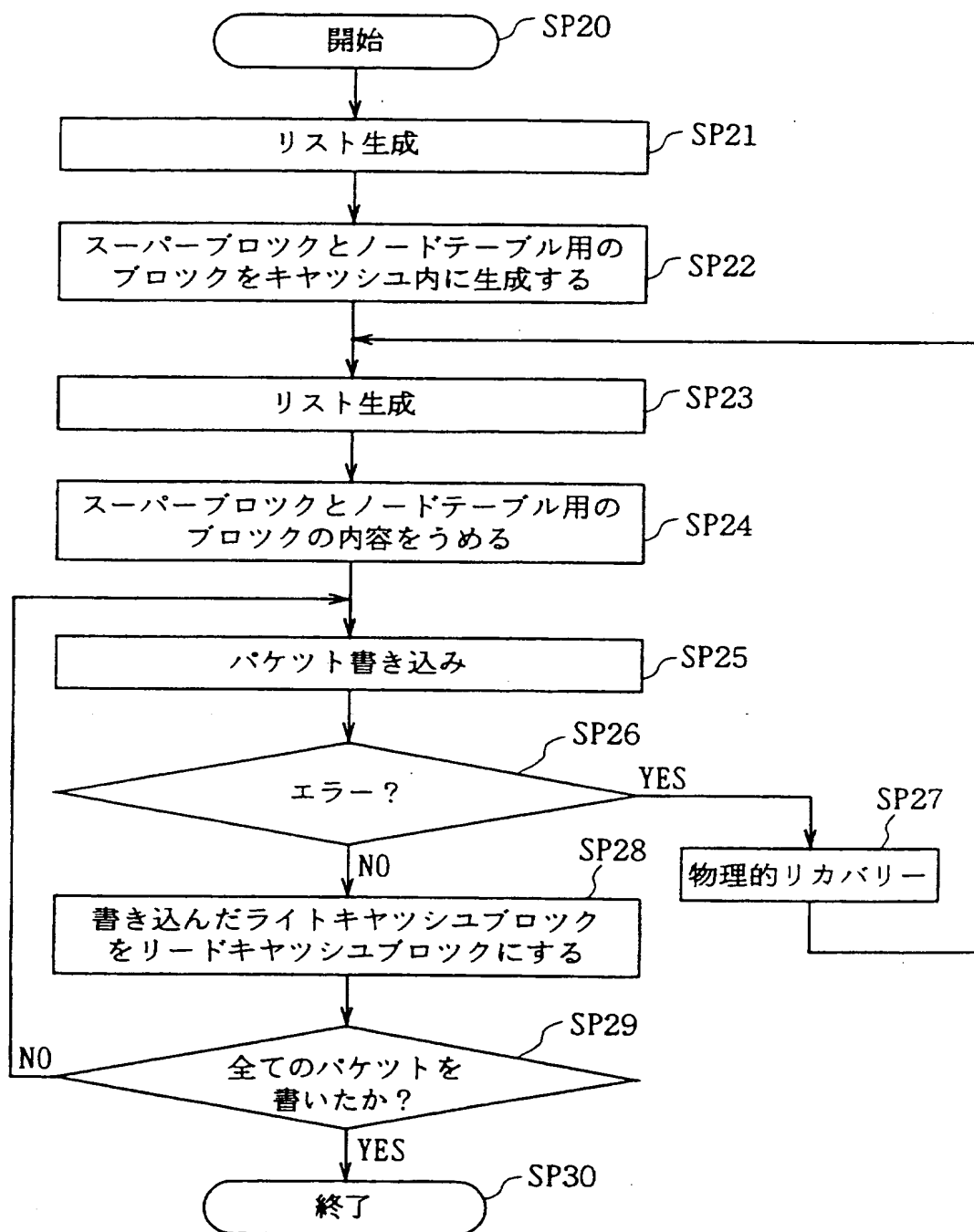


図 29

図30(A)

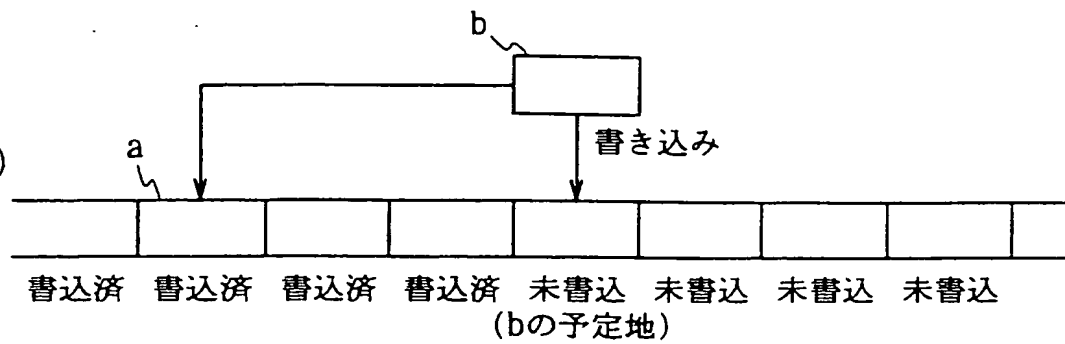


図30(B)

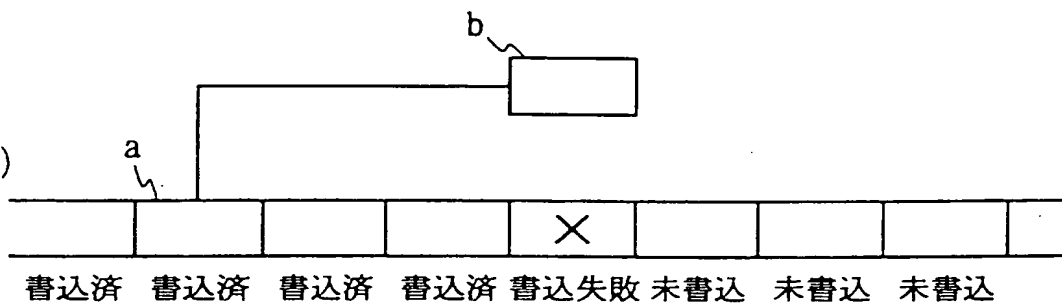
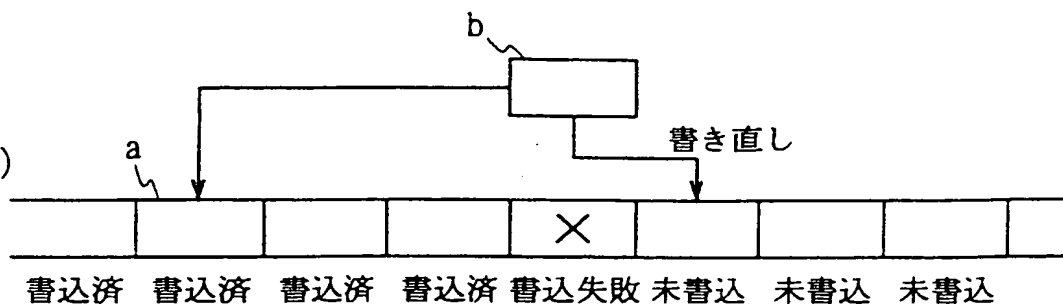


図30(C)



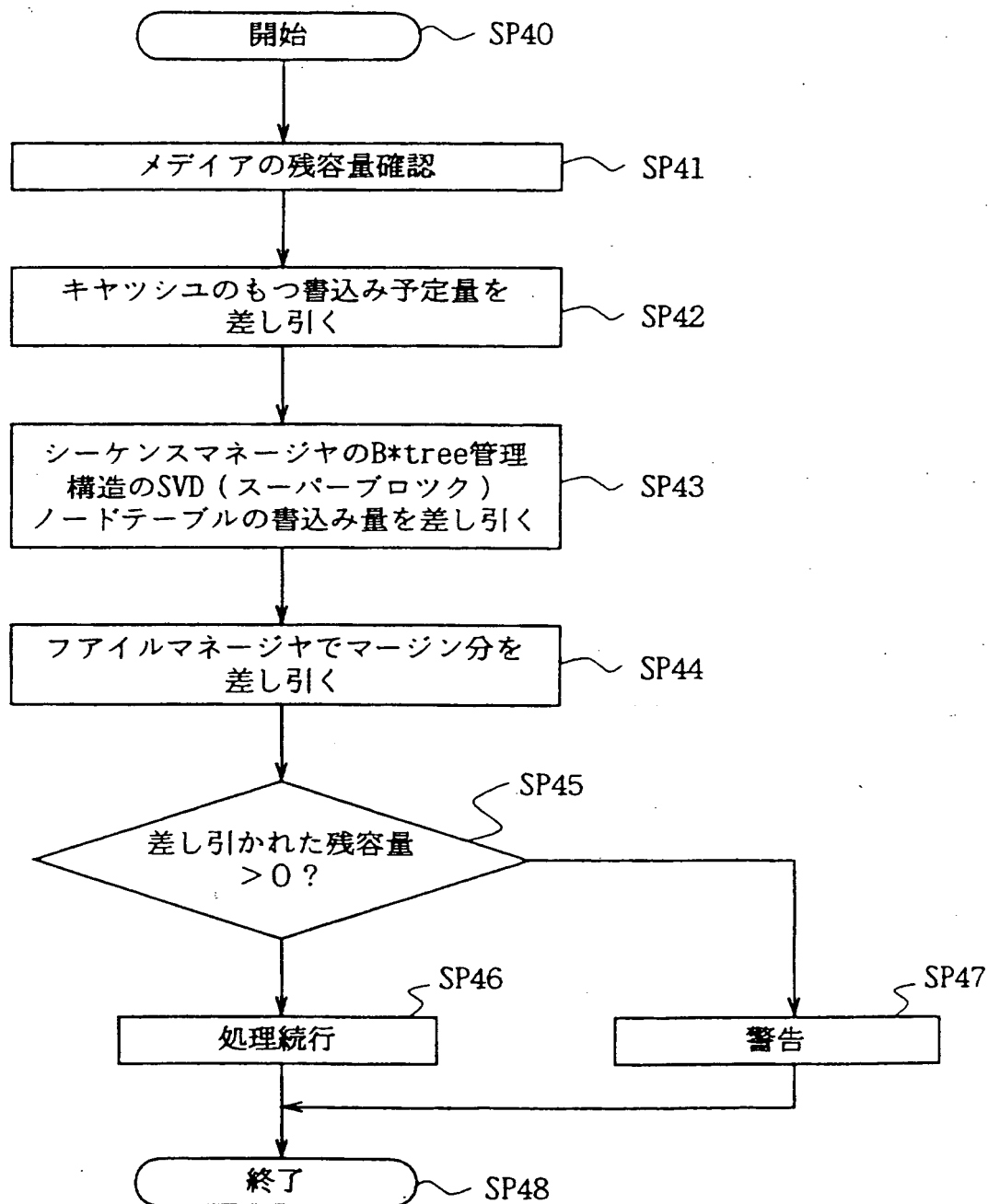


図 31

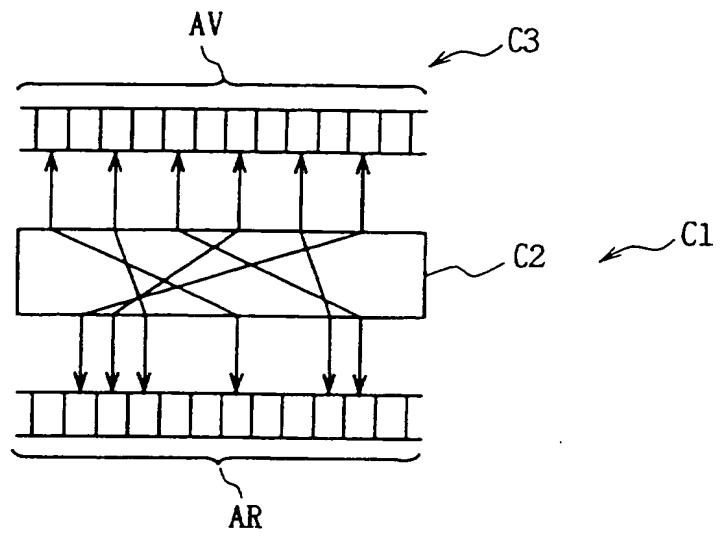


図 32

## 符号の説明

1 ……CD-Rディスク装置、2 ……表示装置、3 ……入力装置、4 ……情報処理装置部、5 ……CD-Rドライバ、6 ……CPU、DISC ……CD-Rディスク、CDRFS ……ファイルシステム、IMM ……仮想デバイスマネージャ、FLM ……ファイルマネージャ、SQM ……シーケンスマネージャ、CAM ……キャツシュマネージャ、SQK ……シーケンスキー、LBA ……物理アドレス。



## INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP96/03194

## A. CLASSIFICATION OF SUBJECT MATTER

Int. Cl<sup>6</sup> G06F3/08

According to International Patent Classification (IPC) or to both national classification and IPC

## B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

Int. Cl<sup>6</sup> G06F3/06-3/08

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Jitsuyo Shinan Koho 1926 - 1996

Kokai Jitsuyo Shinan Koho 1971 - 1996

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

## C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	JP, 6-87229, B2 (Matsushita Graphic Communication Systems, Inc.), November 2, 1994 (02. 11. 94) (Family: none)	1 - 7
Y	JP, 2-214924, A (Hitachi, Ltd.), August 27, 1990 (27. 08. 90) (Family: none)	1 - 7
Y	JP, 2-194455, A (Matsushita Electric Industrial Co., Ltd.), August 1, 1990 (01. 08. 90), Figs. 1 to 2 (Family: none)	1 - 7

☐ Further documents are listed in the continuation of Box C.☐ See patent family annex.

\* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&amp;" document member of the same patent family

Date of the actual completion of the international search

January 27, 1997 (27. 01. 97)

Date of mailing of the international search report

February 4, 1997 (04. 02. 97)

Name and mailing address of the ISA/

Japanese Patent Office

Facsimile No.

Authorized officer

Telephone No.

## 国際調査報告

国際出願番号 PCT/JP96/03194

## A. 発明の属する分野の分類 (国際特許分類 (IPC))

Int. Cl. G 0 6 F 3 / 0 8

## B. 調査を行った分野

調査を行った最小限資料 (国際特許分類 (IPC))

Int. Cl. G 0 6 F 3 / 0 6 - 3 / 0 8

最小限資料以外の資料で調査を行った分野に含まれるもの

日本国実用新案公報 1926 - 1996年  
 日本国公開実用新案公報 1971 - 1996年

国際調査で使用した電子データベース (データベースの名称、調査に使用した用語)

## C. 関連すると認められる文献

引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求の範囲の番号
Y	J P. 6-87229, B2 (松下電送株式会社), 2. 11月. 1994 (02. 11. 94) (ファミリーなし)	1 - 7
Y	J P. 2-214924, A (株式会社日立製作所), 27. 8月. 1990 (27. 08. 90) (ファミリーなし)	1 - 7
Y	J P. 2-194455, A (松下電器産業株式会社), 1. 8月. 1990 (01. 08. 90), 第1-2図 (ファミリーなし)	1 - 7

☐ C欄の続きにも文献が列挙されている。☐ パテントファミリーに関する別紙を参照。

## \* 引用文献のカテゴリー

「A」特に関連のある文献ではなく、一般的技術水準を示すもの  
 「E」先行文献ではあるが、国際出願日以後に公表されたもの  
 「L」優先権主張に疑義を提起する文献又は他の文献の発行日若しくは他の特別な理由を確立するために引用する文献 (理由を付す)  
 「O」口頭による開示、使用、展示等に言及する文献  
 「P」国際出願日前で、かつ優先権の主張の基礎となる出願

の日の後に公表された文献  
 「T」国際出願日又は優先日後に公表された文献であって出願と矛盾するものではなく、発明の原理又は理論の理解のために引用するもの  
 「X」特に関連のある文献であって、当該文献のみで発明の新規性又は進歩性がないと考えられるもの  
 「Y」特に関連のある文献であって、当該文献と他の1以上の文献との、当業者にとって自明である組合せによって進歩性がないと考えられるもの  
 「&」同一パテントファミリー文献

国際調査を完了した日

27. 01. 97

国際調査報告の発送日

04.02.97

国際調査機関の名称及びあて先

日本国特許庁 (ISA/JP)

郵便番号100

東京都千代田区霞が関三丁目4番3号

特許庁審査官 (権限のある職員)

岩崎 伸二

5E 7927

電話番号 03-3581-1101 内線 3520